AD-A111 136 AIR FORCE INST OF TECH WRIGHT-PATTERSON AFB OH SCHOO--ETC F/G 5/8
PREDICTING PILOT OPINION RATINGS OF FLYING QUALITIES OF HIGHLY --ETC(U)
DEC 81 R M ENRIGHT
UNCLASSIFIED AFIT/GAE/AA/80D-3 NL

1 OF 2
AD A
111136

LEVEL II 1

PREDICTING PILOT OPINION RATINGS OF
FLYING QUALITIES OF
HIGHLY CONTROL-AUGMENTED AIRCRAFT
USING AN OPTIMAL PILOT MODEL

by

Randall M. Enright

Capt USAF

7 Dec 1981

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER: AFIT/GAE/AA/80D-3	2. GOVT ACCESSION NO.: AD A111136
3. RECIPIENT'S CATALOG NUMBER	
4. TITLE (and Subtitle): PREDICTING PILOT OPINION RATINGS OF FLYING QUALITIES OF HIGHLY CONTROL-AUGMENTED AIRCRAFT USING AN OPTIMAL PILOT MODEL	5. TYPE OF REPORT & PERIOD COVERED: MS Thesis
	6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s): RANDALL M. ENRIGHT, Capt USAF	8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS: AIR FORCE INSTITUTE OF TECHNOLOGY (AFIT-EN) WRIGHT PATTERSON AFB OH 45433	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS	12. REPORT DATE: 8 Dec 81
	13. NUMBER OF PAGES: 173
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)	15. SECURITY CLASS. (of this report): Unclassified
	15a. DECLASSIFICATION DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved For Public Release; Distribution Unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

Wright-Patterson AFB, OH 45433

...NCH, Major, USAF

...ublic Affairs

18. SUPPLEMENTARY NOTES

28 JAN 1982

Approved for public release; IAW AFR 190-17

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

- Optimal Control
- Optimal Pilot Model
- Pilot Model
- Flying Qualities of Aircraft
- Handling Qualities of Aircraft

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

A computer simulation was constructed using optimal control theory to model the human pilot. The purpose of this study was to determine if this optimal pilot model could be used to evaluate highly control-augmented aircraft and predict the human pilot opinion rating of its flying qualities. Eight aircraft/control system configurations were evaluated twice by the model using two different sets of initial conditions. The performance indices for each of the configurations were plotted and compared to a plot of the human pilot ratings of the same configurations. Variances were noted in the relative positioning of the performance

Block 20 (cont'd)

ratings and the relative positioning of the pilot opinion ratings, and the mean difference in ratings between the performance index predictions and the pilot ratings varied with the initial conditions of the model (the initial conditions of the actual flight tests were unknown). The RMS error in the state response to system noise for each configuration was also plotted and compared to the pilot ratings. The mean difference in ratings between the RMS error predictions, (which were independent of the model initial conditions), and the pilot ratings compared favorably with the mean difference in ratings between two human pilots, though there appeared to be no correlation between the RMS error and the pilots' comments about the aircrafts' flying characteristics. Due to the small sample size involved and the fact that the actual flight evaluations were not designed to compliment the computer simulations, definite conclusions concerning the viability of using an optimal pilot model to predict pilot opinion ratings of highly control-augmented aircraft flying qualities could not be made; however, areas for further investigations were highlighted.

GAE/AA/80D-3

PREDICTING PILOT OPINION RATINGS OF FLYING QUALITIES OF HIGHLY CONTROL-AUGMENTED AIRCRAFT USING AN OPTIMAL PILOT MODEL

THESIS

Presented to the Faculty of the School of Engineering

of the Air Force Institute of Technology

Air University

in Partial Fulfillment of the

Requirements for the Degree of

Master of Science

by

Randall M. Enright

Capt USAF

Graduate Engineering Aeronautics

May 1981

Approved for public release; distribution unlimited.

Preface

The need to find a method of predicting the flying qualities of highly control-augmented aircraft inspired my study of the optimal pilot model, (despite a lack of knowledge of optimal control theory and computer programming). The major portion of the work done on this thesis was concerned with researching optimal control theory and optimal pilot modeling, and in constructing the computer program. Difficulties in obtaining reliable computer service and my reassignment prior to the completion of this study limited the amount of analysis that could be done with the completed optimal pilot model. Still, a foundation has been established for further investigation of this topic and areas for continued study have been identified.

My thanks and appreciation are due to my thesis advisor, Prof. (Capt) James T. Silverthorn, for his expert advice, unending patience, and infinite tolerance of my slow, rambling progress throughout this effort. I would also like to thank Capt Richard Floyd for his help with the use of the Kleinman subroutines.

Randall M. Enright

Table of Contents

List of Figures

List of Symbols

A	Basic aircraft/control system state coefficient matrix
A_1	Aircraft/control/neuromuscular lag system state coefficient matrix
A_o	Aircraft/control/equivalent lag system state coefficient matrix
$\overline{A}$	Closed-loop aircraft/control/equivalent lag system
B	Basic aircraft/control system input coefficient matrix
B_1	Aircraft/control/neuromuscular lag system input coefficient matrix
B_o	Aircraft/control/equivalent lag system input coefficient matrix
C	Displayed variable (instrumentation) coefficient matrix
D	Basic aircraft/control system external disturbance coefficient matrix
D_1	Aircraft/control/neuromuscular lag system external disturbance coefficient matrix
D_2	Aircraft/control/neuromuscular lag system motor noise/external disturbance coefficient matrix
$d(t)$	Function expressing the difference between $f(y)$ and $\hat{f} \cdot \underline{y}(t)$
$E\left\{\cdot\right\}$	Expectation operator
$\hat{e}_1$, $\hat{e}_2$, $\hat{e}_3$	Inertial reference coordinate system
$\hat{e}_{gs}$	Glide slope reference direction
$f(y)$	Non-linear function relating perceived variables to visual threshold
$\hat{f}$	Slope of the linear approximation of $f(y)$
F_{Th}	Force acting on aircraft due to thrust
G	Input rate weighting matrix
g	Gravitational constant
H	Final state weighting matrix
H	Hamiltonian

Δh	Altitude perturbations
h	Altitude
J(u)	Performance index
K	Solution to the optimal control Riccati equation
L	Basic aircraft/control system optimal gain matrix
L_1	Aircraft/control/neuromuscular lag system optimal gain matrix
L_o	Aircraft/control/equivalent lag system optimal gain matrix
ℓ	Element of optimal gain matrix
m	System input order
m_1	Mass of aircraft
n	Basic aircraft/control system order
n_y	Number of displayed variables
P	Estimator error covariance matrix
p(y)	Probability density function of y
Q	Basic aircraft/control system state weighting matrix
Q_1	Aircraft/control/neuromuscular lag system state weighting matrix
Q_o	Aircraft/control/equivalent lag system state weighting matrix
$\mathcal{Q}$	Maximum desired perturbation of state or input
q	Element of state weighting matrix
Δq	Pitch rate perturbations
R	Basic aircraft/control system input weighting matrix
R_1	Aircraft/control/neuromuscular lag system input weighting matrix
R_o	Aircraft/control/equivalent lag system input weighting matrix
Δr	Slant range perturbations
S	Estimator gain matrix
S_o	Sample signal

S_{r_1}	Reconstruction of sample signal using a low sampling rate
S_{r_2}	Reconstruction of sample signal using a high sampling rate
T	Expected value of the difference between $f(y)$ and $\hat{f} \cdot \underline{y}(t)$
$T_{w_g}(s)$	Vertical wind gust shaping filter transfer function
Th	Thrust
Th_{max}	Maximum thrust
t_f	Final time
t_o	Initial time
Δt	Integration time increment
$\underline{u}(t)$	System input vector
$\underline{u}^*(t)$	Optimal input
$\underline{u}_c(t)$	Commanded input
$\dot{\underline{u}}(t)$	System input rate vector
Δu	Forward velocity perturbations
$\underline{u}_o$	Initial input vector
U_o	Nominal forward velocity
$v_y(t)$	Observation noise
V_y	Observation noise covariance matrix
$v_u(t)$	Motor noise
V_u	Motor noise covariance matrix
W	External disturbance covariance matrix
W_1	External disturbance/motor noise covariance matrix
W_o	Nominal downward velocity
$w_g(t)$	External disturbance
$w_{g_1}(t)$	External disturbance/motor noise vector
Δw	Downward velocity perturbations

$\underline{x}(t)$	Basic aircraft/control system state vector
$\tilde{\underline{x}}(t)$	Augmented aircraft/control system state vector
$\hat{\underline{x}}(t)$	Estimated state vector
x_2	Dummy state variable
X	Steady state covariance of state vector $\tilde{\underline{x}}(t)$
$\underline{x}_o$	Initial state vector
$\underline{y}(t)$	Displayed variable vector
$\underline{y}_{p_1}(t)$	Perceived displayed variables (non-linear)
$\underline{y}_{p_2}(t)$	Perceived displayed variables (linear) in terms of the displayed variables/thresholds and the observation noise
$\underline{y}_{p_3}(t)$	Perceived displayed variables (linear) in terms of the displayed variables and the observation noise/thresholds
Y	Steady state covariance of the displayed variables vector $\underline{y}(t)$
Z	Solution to the Lyapunov equation

$$\left.\begin{array}{l} X_u \\ X_w \\ Z_u \\ Z_w \\ Z_{\dot{w}} \\ M_u \\ M_w \\ M_{\dot{w}} \\ M_q \\ X_{\delta_{e_{cs}}} \\ Z_{\delta_{e_{cs}}} \\ M_{\delta_{e_{cs}}} \end{array}\right\} \text{Stability derivatives}$$

$\hat{X}_b$, $\hat{Y}_b$, $\hat{Z}_b$ Body-fixed reference coordinate system

$\Delta\alpha$	Angle of attack perturbations
Γ_0	Nominal glide slope angle
Γ	Input transition matrix
γ	Visual threshold
$\underline{\gamma}$	Small variation in Lagrange multiplier vector
δ_e	Elevator deflection
δe_c	Commanded elevator deflection
δe_{σ}	Elevator deflection command from the control system
δ_{Th_c}	Commanded thrust variation
$\Delta\epsilon$	Angle above glide slope perturbations
ϵ	Arbitrary constant
$\underline{\eta}$	Small variation in input vector
$\Delta\theta$	Pitch attitude perturbations
θ_0	Nominal pitch attitude
λ	Lagrange multiplier
ν	Thrust centerline - Aircraft centerline alignment angle
$\underline{\xi}$	Small variation in state vector
σ_{w_g}	RMS deviation due to wind gusts
σ_u^2	Input variance
σ_y^2	Displayed variable variance
τ	Human pilot internal time delay
τ_n	Neuromuscular time delay
τ_{n_e}	Neuromuscular time delay to elevator
$\tau_{n_{Th}}$	Neuromuscular time delay to thrust
Φ	State transition matrix
Ψ	Displayed variable transition matrix
ω_c	Highest frequency in system response

$\underline{x}^T$	Transpose of $\underline{x}$
$\bar{x}$	Mean value of x

Abstract

A computer simulation was constructed using optimal control theory to model the human pilot. The purpose of this study was to determine if this optimal pilot model could be used to evaluate highly control-augmented aircraft and predict the human pilot opinion rating of its flying qualities. Eight aircraft/control system configurations were evaluated twice by the model using two different sets of initial conditions. The performance indices for each of the configurations were plotted and compared to a plot of the human pilot ratings of the same configurations. Variances were noted in the relative positioning of the performance index ratings and the relative positioning of the pilot opinion ratings, and the mean difference in ratings between the performance index predictions and the pilot ratings varied with the initial conditions of the model (the initial conditions of the actual flight tests were unknown). The RMS error in the state response to system noise for each configuration was also plotted and compared to the pilot ratings. The mean difference in ratings between the RMS error predictions, (which were independent of the model initial conditions), and the pilot ratings compared favorably with the mean difference in ratings between two human pilots, though there appeared to be no correlation between the RMS error and the pilots' comments about the aircrafts' flying characteristics. Due to the small sample size involved and the fact that the actual flight evaluations were not designed to compliment the computer simulations, definite conclusions concerning the viability of using an optimal pilot model to predict pilot opinion ratings of highly control-augmented aircraft flying qualities could not be made; however, areas for further investigations were highlighted.

PREDICTING PILOT OPINION RATINGS OF FLYING QUALITIES OF HIGHLY CONTROL-AUGMENTED AIRCRAFT USING AN OPTIMAL PILOT MODEL

I Introduction

Background

In the design of military aircraft, engineering and flight experience have shown that, for a particular phase of flight, stability and control dynamic characteristics should fall within certain specified limits if an aircraft is to receive a favorable pilot rating of its flying qualities. These limits are contained in the Military Specification - Flying Qualities of Piloted Aircraft, MIL-F-8785B, where dynamic criteria are listed for various levels of flying quality. Aircraft types and the phases of flight are categorized so that for a particular class of aircraft performing a specific phase of flight, the minimum and maximum values of pertinent dynamic characteristics can be found for each flying quality level. Thus, if an aircraft is designed to satisfy the criteria of the highest level of flying quality, one could anticipate favorable pilot opinion ratings during flight tests. While this was generally true in the past, recent increases in the capabilities of fighter aircraft and in the complexity of their control systems have shown that these criteria are no longer always adequate to predict flying qualities.

Problem

A study of highly control-augmented (HCA) aircraft was conducted by Rogers E. Smith and the CALSPAN Advanced Technology Center in

1978 (Ref. 1). As indicated in the report, the YF-17 prototype had been previously simulated in the CALSPAN ground simulator. The aircraft, an HCA system which had been designed to meet the criteria in MIL-F-8785B, demonstrated satisfactory performance in the simulator. Actual flight tests, however, revealed unsatisfactory behavior during the landing phase and the pilot's rating of the YF-17 for this task was the worst possible on the Cooper-Harper Pilot Opinion Rating Scale (Fig. 1). The USAF/CALSPAN variable stability NT-33A aircraft was flown using several HCA systems to gather data on their longitudinal flying qualities. Considerable discrepancies were noted when pilot ratings of the landing approach task were compared to the flying quality levels predicted by MIL-F-8785B. The report concluded that the specifications, being based solely on classical aircraft characteristics, were not applicable to aircraft with highly augmented control systems. Predicting the flying qualities of these aircraft poses a problem. A method for anticipating aircraft flying qualities that is applicable to HCA aircraft is needed.

Objectives

The objectives of this thesis were to construct an optimal pilot computer program and evaluate the CALSPAN aircraft/control system configurations performing an Instrument Landing System (ILS) approach. A correlation between the resulting performance index from the computer simulation and the pilots' ratings of the actual flight tests would indicate the feasibility of predicting the flying qualities of HCA aircraft using optimal control theory.

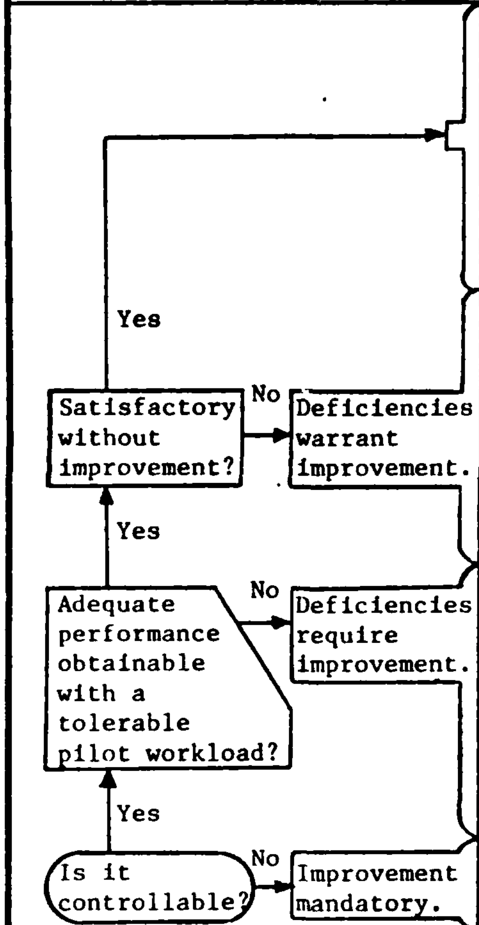
ADEQUACY FOR SELECTED TASK OR REQUIRED OPERATION	AIRCRAFT CHARACTERISTICS	DEMANDS ON THE PILOT IN SELECTED TASK OR REQUIRED OPERATION	PILOT RATING	FLYING QUALITY LEVELS
	Excellent. Highly desirable.	Pilot compensation not a factor for desired performance.	1	1
	Good. Negligible deficiencies.	Pilot compensation not a factor for desired performance.	2	
	Fair. Some mildly unpleasant deficiencies.	Minimal pilot compensation required for desired performance.	3	
	Minor but annoying deficiencies.	Desired performance requires moderate pilot compensation.	4	2
	Moderately objectionable deficiencies.	Adequate performance requires considerable pilot compensation.	5	
	Very objectionable but tolerable deficiencies.	Adequate performance requires extensive pilot compensation.	6	
	Major deficiencies.	Adequate performance unattainable with maximum tolerable pilot compensation. Controllability not in question.	7	3
	Major deficiencies.	Considerable pilot compensation is required for control.	8	
	Major deficiencies.	Intense pilot compensation is required to retain control.	9	
	Major deficiencies.	Control will be lost during some portion of required operation.	10	

Figure 1 Cooper-Harper Pilot Opinion Rating Scale

Approach

This thesis is presented in seven parts. First, Chapter I provides a background on the prediction of flying qualities and introduces the problem of applying the prediction method to HCA aircraft. Chapter II discusses the optimal control system and explains how it is expanded to model the human pilot. Chapter III details the modeling of the aircraft system. In Chapter IV, the details of the construction of the optimal pilot computer program, OPSACT, is discussed. Chapter V presents the results of the evaluation of the CALSPAN aircraft/control system configurations by OPSACT, and Chapter VI contains the conclusions and recommendations of this thesis. The Appendices contain detailed mathematical developments of optimal control theory and the optimal pilot, an OPSACT program listing, and the CALSPAN aircraft/control system configuration data.

II Modeling the Pilot

Optimal Control System

The purpose of the optimal control system is to find the control input $\underline{u}^*(t)$ that minimizes the performance index

$$J(\underline{u}) = f_1[\underline{x}(t_f)] + \int_{t_o}^{t_f} f_2[\underline{x}(t),\underline{u}(t),t]dt \tag{1}$$

subject to the constraining equation

$$\dot{\underline{x}}(t) = f_3[\underline{x}(t),\underline{u}(t),t] \; ; \quad \underline{x}(t_o) = \underline{x}_o \tag{2}$$

The performance index is an arbitrary mathematical expression designed to measure how well a system is performing a particular task. For an aircraft making an ILS approach, where both positive and negative deviations from the glide slope are assumed equally undesirable, measurement of the mean square error of the states and the inputs from their nominal values is an accurate means of indicating how well the aircraft is following the glide slope. Thus, for the flight task this thesis investigated, the performance index was

$$\begin{aligned} J(\underline{u}) &= \tfrac{1}{2}\, \underline{x}^T(t_f)\; H\; \underline{x}(t_f) \\ &+ \tfrac{1}{2} \int_{t_o}^{t_f} [\underline{x}^T(t)Q\underline{x}(t) + \underline{u}^T(t)R\underline{u}(t)]dt \end{aligned} \tag{3}$$

The square matrices H and Q are symmetric, positive semi-definite weighting matrices and R is a symmetric, positive definite weighting matrix.

These weighting matrices express the relative importance of the deviations of each state variable and control input. They may be expressed as functions of time. For example, as an aircraft nears the ground during an approach, deviations in altitude become more important relative to other variables. The element of the weighting matrix Q corresponding to the altitude perturbation state variable should therefore increase with time. Increased weighting indicates to the optimal control system that a larger "penalty" is associated with deviations from the nominal altitude. In order to minimize the value of the performance index, the control system would place more emphasis on controlling deviations in altitude than on other states or inputs. However, this does not mean that altitude perturbations or any other state will be at its nominal value at the final time. The first term on the right-hand side of Eq. 3 attempts to force some or all of the states to their nominal values at the final time. This thesis, however, investigated only the approach task and did not attempt to model the transition from the ILS glide slope to the landing. The aircraft's task was viewed as maintaining an infinitely long glide slope, thus the weighting matrices were constant with respect to time and the first term on the right-hand side of Eq. 3 was dropped.

It was assumed that the aircraft system could be represented by a linear, time invariant expression. The constraining equation became

$$\underline{\dot{x}}(t) = A\underline{x}(t) + B\underline{u}(t) \; ; \quad \underline{x}(t_o) = \underline{x}_o \tag{4}$$

Applying the First Necessary Conditions for Optimality, (as developed in Appendix A), the optimal input was

$$\underline{u}^*(t) = -L(t)\underline{x}(t) = -R^{-1}B^TK(t)\underline{x}(t) \qquad (5)$$

where K(t) satisfied the equation

$$-\dot{K}(t) = K(t)A + A^TK(t) + Q - K(t)BR^{-1}B^TK(t) \; ;$$

$$K(t_f) = H = 0 \qquad (6)$$

Since the final time of the task is far off into the future relative to the system time constants, the solution to Eq. 6 was constant with respect to time. Thus Eq. 6 became

$$0 = KA + A^TK + Q - KBR^{-1}B^TK \qquad (7)$$

and Eq. 5 became

$$\underline{u}^*(t) = -L\underline{x}(t) = -R^{-1}B^TK\underline{x}(t) \qquad (8)$$

Feedback of $\underline{u}^*(t)$ resulted in a linear, deterministic optimal control system (Fig. 2).

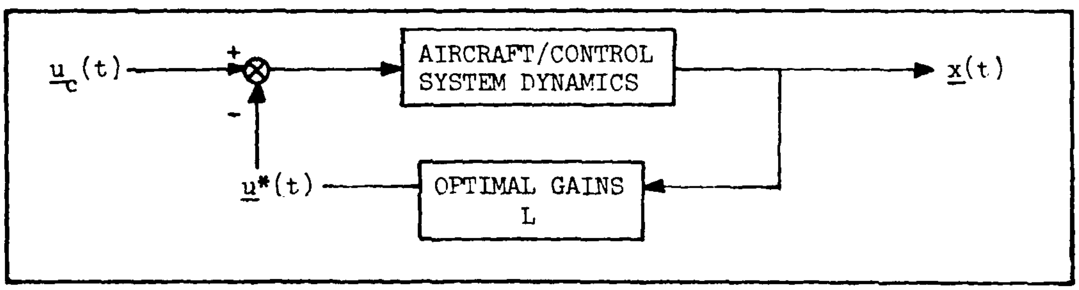


Figure 2 Optimal Control System

Optimal Pilot Model

In 1970, Anderson stated that, given a particular aircraft and task, the "optimal" pilot will adapt a control stategy that minimizes his numerical opinion rating (Ref. 2). He suggested that combining the optimal pilot control strategy with modern control theory would yield a method of evaluating aircraft performance that would take into account the pilot and his rating of the aircraft dynamics. This combination is reflected in the optimal pilot model. This model evolves from the optimal control system through the inclusion of features that model human response characteristics and the effects of random errors.

Human Response Characteristics. The pilot has no means of directly observing the system states. He receives state information from the cockpit instrumentation and his perceptions of the world around him (peripheral vision cues, aircraft accelerations). If all information is assumed to come from the instrumentation, (as was the case for this study), then the pilot must estimate the states based on the cockpit display. This estimation process is modeled by a Kalman filter that reconstructs the states from the information available from the instrumentation. A detailed mathematical development of the estimation process is contained in Appendix B. Inclusion of the cockpit instrumentation and the estimation process is depicted in Fig. 3.

Human response includes various internal time delays. These delays, usually due to lags in relaying and processing visual information in the brain, are modeled by lumping them into a single internal time delay, τ. Research has shown that a time delay of 0.2 seconds is typical (Refs. 3,4,5). The pilot, realizing that he cannot act instantaneously upon receipt of displayed information, attempts to predict the

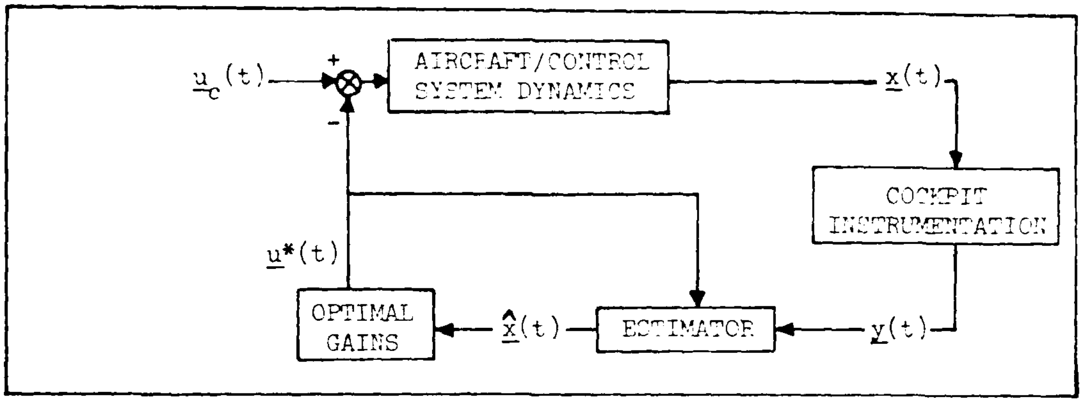


Figure 3 Modeling Cockpit Instrumentation and the Estimation Process

states of the aircraft and compensate for his inherent time delay. This prediction process is developed in Appendix B and is included in the model as indicated in Fig. 4 (Ref 6).

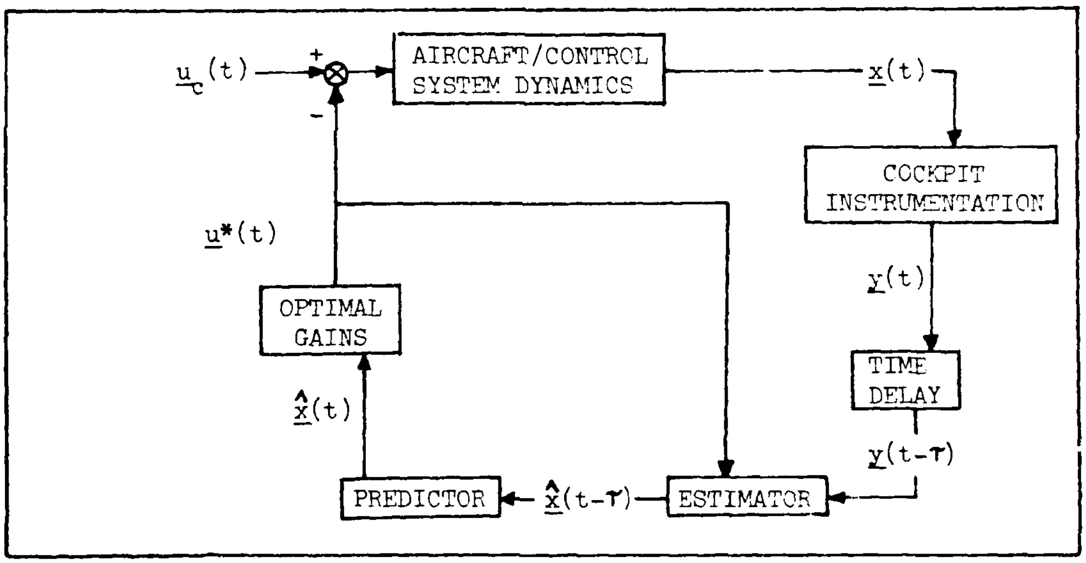


Figure 4 Modeling Time Delay and Prediction Process

Another delay in human response is due to the physiological limitation on the rate at which the pilot can effect the controls. This limitation, usually defined as a neuromuscular lag, is modeled by including a control rate term in the performance index. Thus, Eq. 3 becomes

$$J(\underline{u}) = \frac{1}{2} \int_{t_0}^{t_\infty} \Big[\underline{x}^T(t) Q \underline{x}(t) + \underline{u}^T(t) R \underline{u}(t) + \dot{\underline{u}}^T(t) G \dot{\underline{u}}(t) \Big] dt \qquad (9)$$

where Q and R are now symmetric, positive semi-definite weighting matrices and G is a symmetric, positive definite weighting matrix. Rynaski and Whitbeck have shown that adding a control rate term to the performance index has the same effect as including a first-order lag of the form $\frac{1}{\tau_n + 1}$ in the aircraft model (Ref. 8), where τ_n is the neuromuscular time constant. However, if the neuromuscular lag is modeled simply by inserting the first-order lag, the input from the optimal pilot to the aircraft is calculated incorrectly. This is shown below. The system with the first-order lag and an optimal feedback of the aircraft states ($-L\underline{x}$) is depicted in Figure 5a. This is the form of the solution that is desired. The state equations become

$$\dot{\tilde{\underline{x}}}(t) = A_1 \tilde{\underline{x}}(t) + B_1 \underline{u}^*(t) \qquad (10)$$

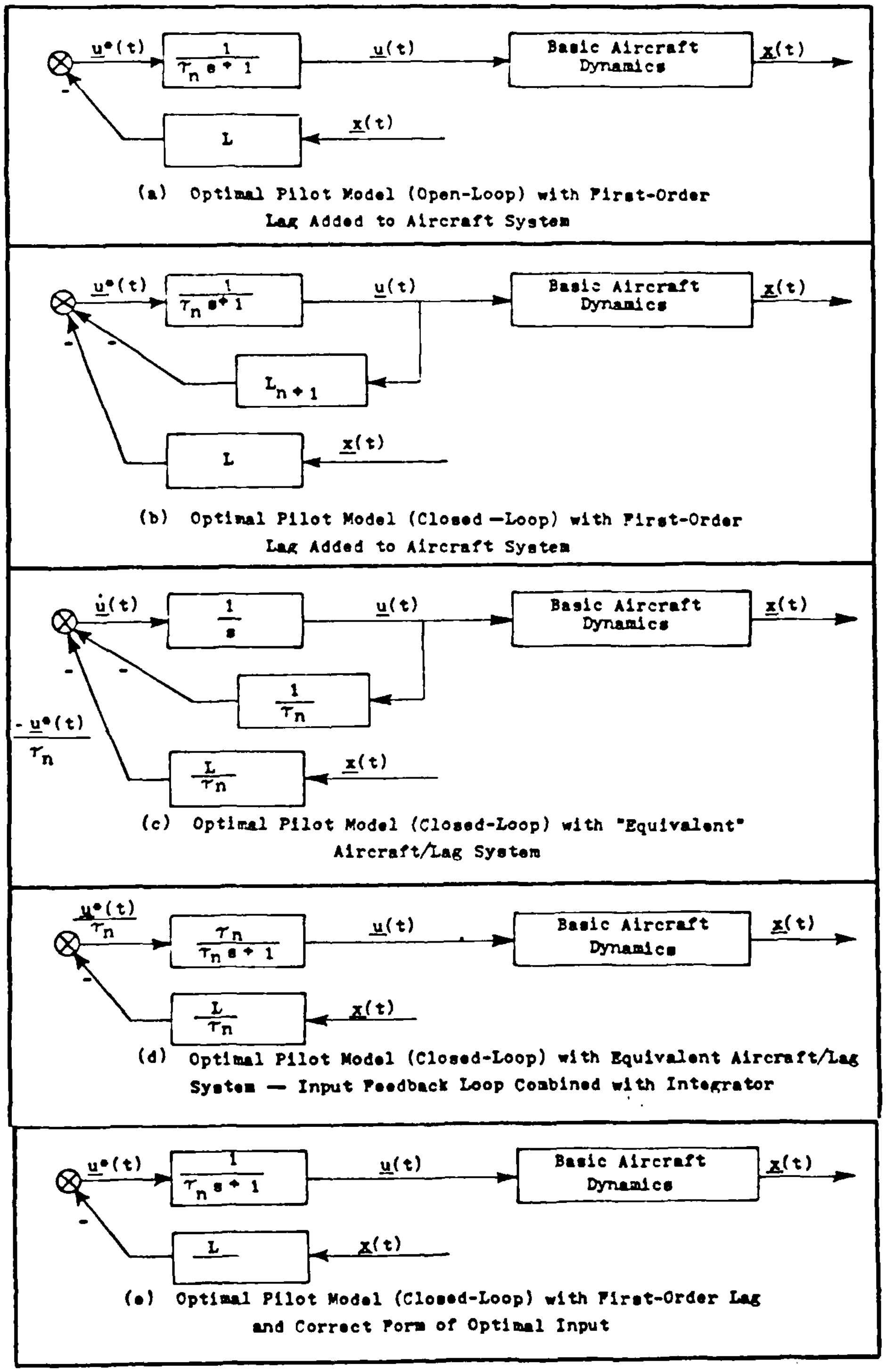
where

$$\tilde{\underline{x}}(t) = \begin{bmatrix} \underline{x}(t) \\ \hline \underline{u}(t) \end{bmatrix}, \quad A_1 = \left[\begin{array}{c|c} A & B \\ \hline 0 & \frac{1}{-\tau_n} \end{array}\right], \text{ and } B_1 = \begin{bmatrix} 0 \\ \hline \frac{1}{\tau_n} \end{bmatrix} \qquad (11)$$

The optimal input is

$$\underline{u}^*(t) = -L\underline{x}(t) = \tau_n \dot{\underline{u}}(t) + \underline{u}(t) \qquad (12)$$

If the augmented state equation given by Eqn. 10 were to be used in conjunction with optimal control theory, the resulting optimal



(a) Optimal Pilot Model (Open-Loop) with First-Order Lag Added to Aircraft System

(b) Optimal Pilot Model (Closed —Loop) with First-Order Lag Added to Aircraft System

(c) Optimal Pilot Model (Closed-Loop) with "Equivalent" Aircraft/Lag System

(d) Optimal Pilot Model (Closed-Loop) with Equivalent Aircraft/Lag System — Input Feedback Loop Combined with Integrator

(e) Optimal Pilot Model (Closed-Loop) with First-Order Lag and Correct Form of Optimal Input

Figure 5 Modeling the Neuromuscular Lag

control $\underline{u}^*(t)$ would be as shown in Figure 5b. Notice that $\underline{u}^*(t)$ would include not only the aircraft states but also feedback of input variable $\underline{u}(t)$. This L_{n+1} feedback term would effectively modify the time constant of the first-order lag. To avoid this, the following modification is used.

In order to achieve the desired form of the closed-loop optimal input, the aircraft/lag system is modeled as if $\dot{\underline{u}}(t)$ was the input from the optimal pilot. This is done by rearranging the expression for the open-loop input (Eqn. 12), which yields

$$\dot{\underline{u}}(t) = \frac{1}{\tau_n}\underline{u}^*(t) - \frac{1}{\tau_n}\underline{u}(t) = \frac{-L}{\tau_n}\underline{x}(t) - \frac{1}{\tau_n}\underline{u}(t) \qquad (13)$$

The remodeled system is depicted in Figure 5c. It should be noted that the system in Fig. 5a is equivalent to the system in Fig. 5c. The open-loop state equation now becomes

$$\dot{\tilde{\underline{x}}}(t) = A_o\tilde{\underline{x}}(t) + B_o\dot{\underline{u}}(t) \qquad (14)$$

$$\text{where} \quad A_o = \left[\begin{array}{c|c} A & B \\ \hline 0 & 0 \end{array}\right] \quad \text{and} \quad B_o = \left[\begin{array}{c} 0 \\ \hline 1 \end{array}\right] \qquad (15)$$

The closed-loop gain matrix is

$$L_o = \left[\begin{array}{c|c} \frac{L}{\tau_n} & \frac{1}{\tau_n} \end{array}\right] \qquad (16)$$

which is calculated from

$$\underline{\dot{u}}(t) = -L_o\underline{\tilde{x}}(t) , \quad L_o = R_o^{-1}B_o^TK_o \tag{17}$$

where

$$0 = K_oA_o + A_o^TK_o + Q_o - K_oB_oR_o^{-1}B_o^TK_o \tag{18}$$

and where $Q_o = \left[\begin{array}{c|c} Q & 0 \\ \hline 0 & R \end{array}\right]$, $R_o = G$ (where G is defined in Eqn. 9).

The input feedback loop and the integrator in Fig. 5c can be combined to yield the system depicted in Fig. 5d, which is equivalent to the system in Fig. 5e. From this figure it can be seen that the correct feedback gain for the optimal pilot model with a first-order lag is calculated using Eqns. 14 through 18 and then dropping the n+i, i = 1,2,...m, terms from L_o. Thus, the optimal feedback gain matrix for the aircraft/control system with the neuromuscular lag is

$$L_1 = \text{first } n \text{ terms of } L_o = \left[\frac{L}{\tau_n}\right]$$

Therefore, using Eqns. 14 - 18 to determine the feedback gain produces a closed-loop system as shown in Fig. 5e. The desired value of τ_n is obtained by adjusting the values of the weighting matrices. For a given submatrix R (which is now part of the state weighting

matrix Q_o, where $Q_o = \left[\begin{array}{c|c} Q & 0 \\ \hline 0 & R \end{array}\right]$), there is a one-to-one relationship between

R_o = G and τ_n (Ref. 6). τ_n is determined through an iterative process of guessing a value for R_o, calculating the gain matrix L_o, comparing the reciprocal of the n+i; i = 1,2,...m. terms of L_o with the desired values of τ_n, and then reguessing R_o. If there is more than one input, $u_1^*(t)$ and $u_2^*(t)$ for example, the optimal gain matrix will be of the form

$$L_o = \left[\begin{array}{cccc|cc} \ell_{1,1} & \ell_{1,2} & \cdots & \ell_{1,n} & \ell_{1,n+1} & \ell_{1,n+2} \\ \ell_{2,1} & \ell_{2,2} & \cdots & \ell_{2,n} & \ell_{2,n+1} & \ell_{2,n+2} \end{array}\right] \tag{19}$$

where $\ell_{1,n+1}$ corresponds to $u_1^*(t)$ and $\ell_{2,n+2}$ corresponds to $u_2^*(t)$. In researching this subject, no information could be found concerning the consequences of having more than one input. The significance of the optimal gains $\ell_{1,n+2}$ and $\ell_{2,n+1}$ and the results of varying the off-diagonal terms of R_o was explored briefly in this thesis and will be discussed in Chapter IV. More importantly, the significance of the weighting term in the R submatrix for the new state $u_1^*(t)$ and its effects on the optimal gain calculations for all other new states $u_i^*(t)$; i = 2,3,...m, was found to be critical in this thesis and will also be discussed later.

Human limitations also include an effect called "remnant", (Ref. 10), defined in this thesis to be the inherent random human errors in the observation of the instruments and in the execution of desired control inputs. The final human limitation modeled in this thesis is "threshold",

which refers to the pilot's inability to detect, or his indifference to, small changes in the instrument gauges. The modeling of remnant and threshold is combined with the modeling of the effects of random external disturbances, which is discussed next.

Random Input Effects. Random inputs to the system are due to external disturbances, such as wind gusts, and pilot errors. In investigating longitudinal handling qualities during an ILS landing task, only vertical wind gusts are considered to have a significant effect. Including these vertical wind gusts in the optimal pilot model will alter the aircraft/control system dynamic equations. Eq. 10 becomes

$$\dot{\underline{\tilde{x}}}(t) = A_1\underline{\tilde{x}}(t) + B_1\underline{u}^*(t) + D_1 w_g(t) \tag{20}$$

where $D_1 = \begin{bmatrix} D \\ \hline 0 \end{bmatrix}$ and $w_g(t)$ represents random vertical wind gusts.

The D matrix is determined by realizing that the state Δw in the system equations of motion, which are discussed in more detail in Chapter III, is comprised of the downward velocity with respect to the ground, Δw, minus the movement of the air mass w_g. For example, the equation of motion describing perturbations in forward velocity is sometimes written as

$$\Delta\dot{u} = \ \ldots + X_w(\Delta w) + \ldots$$

Actually, this equation should be

$$\Delta\dot{u} = \ldots + X_w(\Delta w - w_g) + \ldots = \ldots + X_w\Delta w - X_w w_g + \ldots$$

Since vertical wind gusts significantly affect the Δu, Δw, and Δq states, the D matrix is defined as

$$D = \begin{bmatrix} -X_w \\ -Z_w \\ -M_w \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

where X_w, Z_w, and M_w are the dimensional stability derivatives associated with Δw in the system equations of motion.

Random wind gusts can be modeled by passing white noise through a shaping filter designed to produce random outputs with a power spectral density similar to that of actual verticle wind gusts.

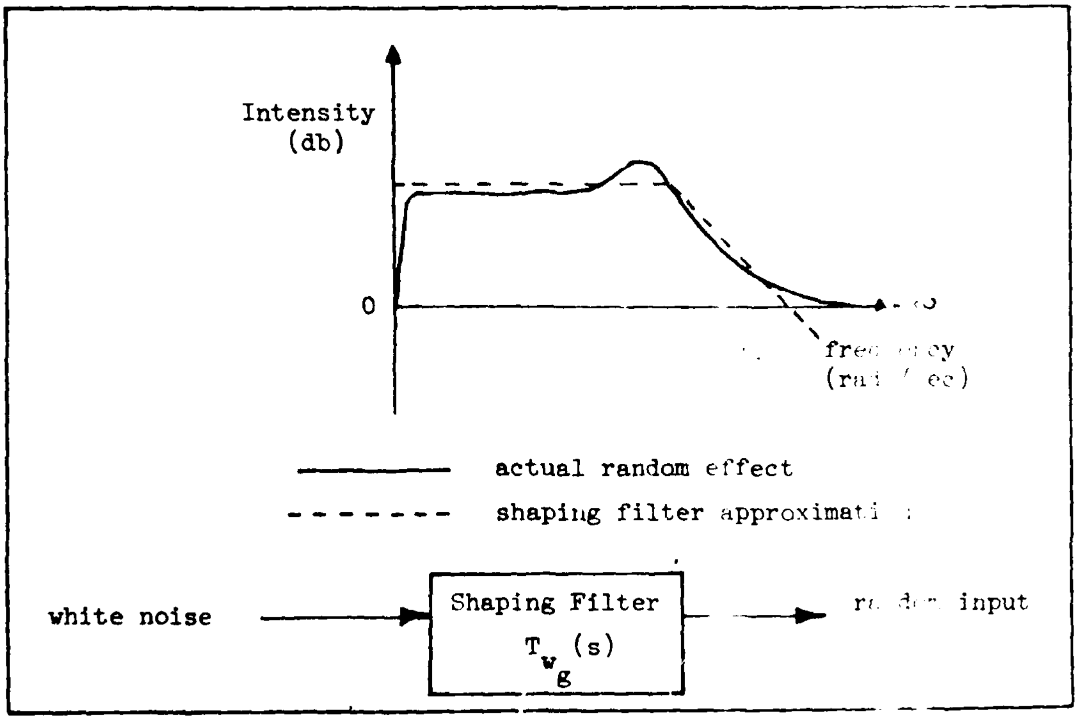


Figure 6 Modeling a Random Effect

Figure 6 demonstrates the use of white noise and a shaping filter to approximate a random effect. The shaping filter for verticle wind gusts at the altitudes for an ILS landing approach is specified in the military specification MIL-F-8785B as

$$T_{w_g}(s) = \frac{\sigma_w \sqrt{\frac{h_o}{U_o}} \left[1 + \frac{\sqrt{3}\, h_o}{U_o} s \right]}{\left[1 + \frac{h_o}{U_o} s \right]^2} \qquad (21)$$

where h_o is the nominal ILS altitude and U_o is the nominal forward velocity. For this thesis, the wind gust is approximated for an altitude of 1900 ft and a velocity of 205 ft/sec. MIL-F-8785B specifies σ_w at these conditions to be 6 ft/sec. In this study, the vertical wind gusts were modeled as white noise. This approximation was used to avoid introducing additional state variables to account for shaping filters. This avoided the computational complexities of a large number of variables. From Eq. 21, the power spectral density of the vertical wind gusts is flat with an amplitude given by

$$T_{w_g}(s) \Big|_{s=0} = 18.3$$

Thus, the intensity of the gusts is

$$W = (18.3)^2 = 334$$

Human controller remnant, as defined by Levison, Baron, and Kleinman (Ref. 10), is the portion of the pilot's input not associated with the aircraft/

control system input. Though vertical wind gusts are normally included as part of the system dynamic equations, this thesis models them along with observation and motor noise as a separate input source.

Motor noise consists of the neuromuscular errors the pilot makes in activating the controls and accounts for the fact that he does not have perfect knowledge of his control inputs. In modeling the effects of motor noise, it should be recalled that though the aircraft/control system had to be modeled according to Eq. 14 in order to generate the appropriate feedback gains, the system still has the same control structure and is modeled according to Eq. 10 (Ref. 6). By including the motor noise, the control input defined by Eq. 8 becomes

$$\underline{u}^*(t) + v_u(t) = -L\hat{\underline{x}}(t) + v_u(t) = \tau_n\dot{\underline{u}}(t) + \underline{u}(t) \tag{22}$$

or equivalently

$$\dot{\underline{u}}(t) = \frac{-L}{\tau_n}\hat{\underline{x}}(t) + \frac{v_u(t)}{\tau_n} - \frac{\underline{u}(t)}{\tau_n} \tag{23}$$

where $v_u(t)$ represents the motor noise. The system is now modeled by including the motor noise into Eq. 20

$$\dot{\tilde{\underline{x}}}(t) = A_1\tilde{\underline{x}}(t) + B_1\underline{u}^*(t) + D_2 w_{g_1}(t) \tag{24}$$

where $w_{g_1}(t) = \begin{bmatrix} w_g(t) \\ \hline \dfrac{v_u(t)}{\tau_n} \end{bmatrix}$ and $D_2 = \left[\begin{array}{c|c} D_1 & 0 \\ \hline 0 & 1 \end{array}\right]$

Studies of human remnant have shown that motor noise is proportional to the variance of the inputs (Ref. 10). Specifically,

$$V_u = .003\pi \, E\left\{[\underline{u}^*(t)-\overline{\underline{u}}^*(t)][\underline{u}^*(t)-\overline{\underline{u}}^*(t)]^T\right\} \tag{25}$$

Also, the intensity of the external disturbance/motor noise is

$$W_1 = D_2 \, E\left\{w_{g_1}(t) w_{g_1}{}^T(t)\right\} D_2$$

$$= \left[\begin{array}{c|c} D_1 W D_1^T & 0 \\ \hline 0 & \dfrac{V_u}{\tau_n{}^2} \end{array}\right] \tag{26}$$

The determination of the motor noise matrix V_u is done in an iterative manner. A guess for V_u must be made in order to form W_1. The estimator gain is then calculated (Appendix B). The steady state covariance of the state vector $\underline{x}(t)$, which contains the variance of the subvector $\underline{u}^*(t)$, is then calculated using a relationship developed by Kleinman (Ref. 11). This relation is

$$X = \left\{(\underline{x}-\overline{\underline{x}})(\underline{x}-\overline{\underline{x}})^T\right\} = e^{A_1\tau} P \, e^{A_1^T\tau} + \int_0^{\tau} [e^{A_1\sigma} W_1 \, e^{A_1^T\sigma}] \, d\sigma$$

$$+ \int_0^{\infty} [e^{\overline{A}\sigma} e^{A_1\tau} \, PC^T V_y^{-1} CP \, e^{A_1^T\tau} e^{\overline{A}\sigma}] d\sigma \tag{27}$$

where $\overline{A} = A_o - B_o L_o$, P is the estimator error covariance matrix (to be discussed later), and V_y is the observation noise matrix (to be discussed next). The $n^{th} + 1$ element on the principle diagonal of the X matrix,

(corresponding to the $n^{th}+1$ state, $u_i^*(t)$; $i = 1$), is the variance of the input, σ_u^2. The first guess of $V_{u_{i,i}}$ is now compared to $.003\pi\sigma^2_{u_{n+i}}$, If the difference is greater than some arbitrary tolerance, (5% of $.003\pi\sigma^2_{u_{n+i}}$ was used for this thesis), then the new guess of $V_{u_{i,i}}$ is set equal to $.003\pi\sigma^2_{u_{n+i}}$ and the procedure is repeated. This process is described in more detail in Chapter IV.

The final component of remnant modeled in this thesis is the observation noise. This effect accounts for the errors the pilot makes in reading the instruments and the random errors in the instruments themselves. Similar to the motor noise, the observation noise is proportional to the variance of the displayed variables. Specifically,

$$V_y = .01\pi\, E\left\{[\underline{y}(t)-\overline{\underline{y}}(t)][\underline{y}(t)-\overline{\underline{y}}(t)]^T\right\} \tag{28}$$

Determination of V_y is done in the same iterative manner as with V_u, noting that

$$E\left\{[\underline{y}(t)-\overline{\underline{y}}(t)][\underline{y}(t)-\overline{\underline{y}}(t)]^T\right\}$$

$$= C\ \ E\left\{[\underline{x}(t)-\overline{\underline{x}}(t)][\underline{x}(t)-\overline{\underline{x}}(t)]^T\right\} C^T \tag{29}$$

where C is the displayed variable (instrumentation) coefficient matrix.

One other item is included in the determination of the observation noise matrix. When reading the instrument gauges, the pilot cannot detect changes in the readings that are smaller than a certain threshold value. Also, the pilot may choose to ignore readings that he can see but are smaller than his indifference threshold. Since the

indifference threshold differs from pilot to pilot, it was assumed that the optimal pilot would react to any changes in the gauges that he could detect. The visual threshold was modeled assuming that the pilot looks directly at the instrument panel. Studies have shown that when viewing an object straight on, the human eye can detect changes in position equivalent to .05° of visual arc and changes in rates equivalent to .18° of arc per second (Ref. 3). For example, if a vertical velocity indicator was 3 inches in diameter and was calibrated as shown in Fig. 7, a movement of the indicator of .024 inches would correspond to 1 ft/sec in vertical velocity.

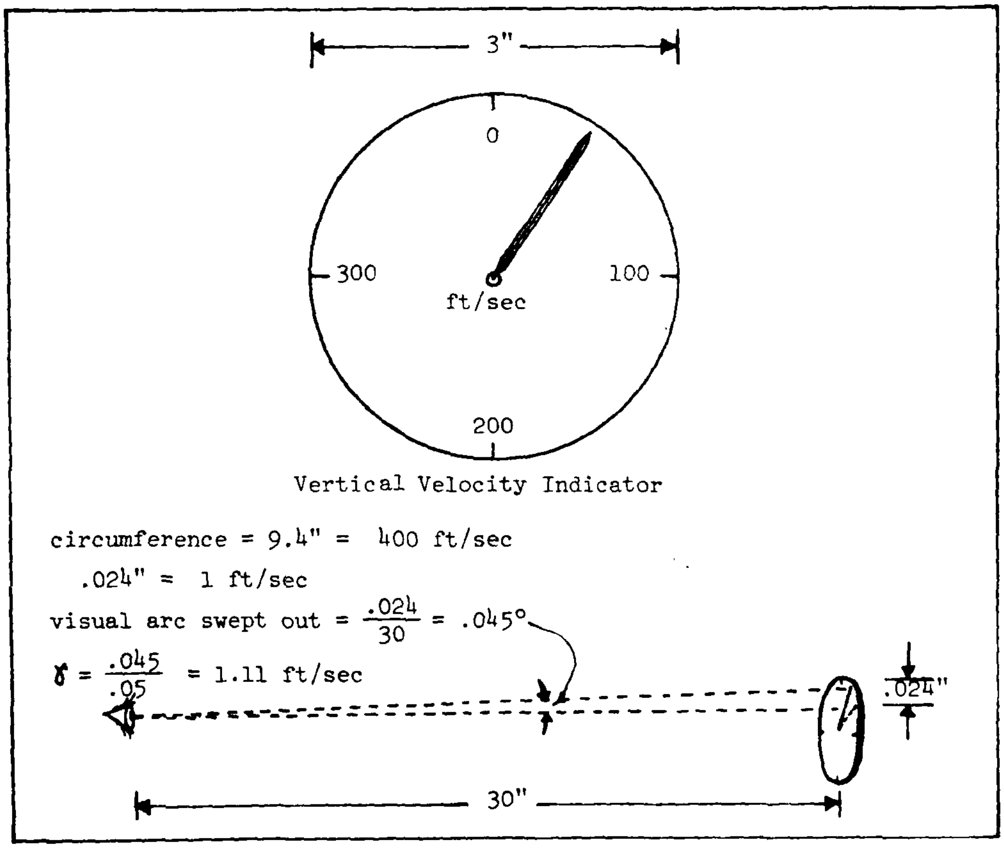


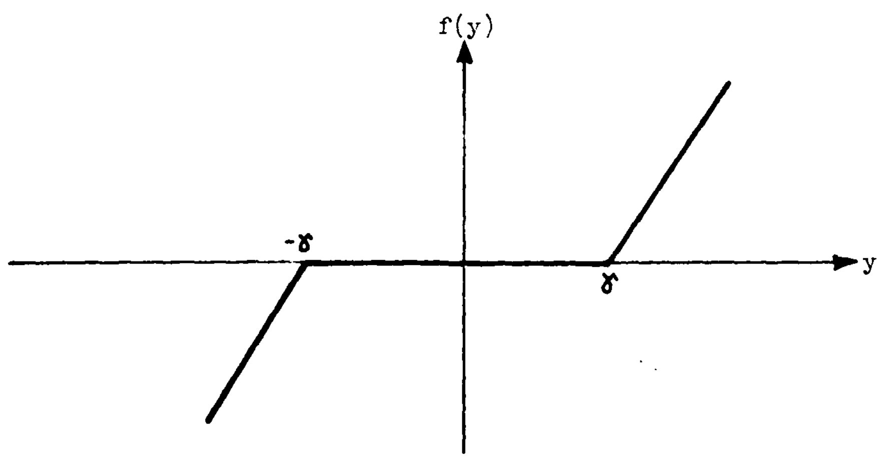
Figure 7 Determination of Visual Threshold

Assuming the instrument was 30 inches away from the pilot, the visual arc swept out by the .024 inch movement of the indicator is .045°/ft/sec. Dividing the minimum visual arc perceivable by the actual arc swept out results in the visual threshold γ. In this example, $\gamma = 1.11$ ft/sec. Therefore, any vertical velocity indication less than 1.11 ft/sec would not be detected by the pilot.

The information the pilot is able to perceive $y_{p_1}(t)$ is now described by the non-linear function $f(y)$ and the observation noise $v_y(t)$.

$$y_{p_1}(t) = f(y) + v_y(t) \tag{30}$$

The function $f(y)$ is depicted in Fig. 8.



$$f(y) = \begin{cases} y-\gamma & \text{for } y \geq \gamma \\ 0 & \text{for } -\gamma < y < \gamma \\ y+\gamma & \text{for } y \leq -\gamma \end{cases}$$

Figure 8 Threshold Function f(y)

In order to incorporate the threshold effect into the pilot model, a linear representation of f(y) is needed. This representation is developed by Kleinman (Ref. 3). A difference function d(t) is defined as follows:

$$d(t) = f(y) - \hat{f} \cdot y(t) \tag{31}$$

where $\hat{f} \cdot y$ is the linear approximation of f(y) and minimizes the relation

$$\begin{aligned} T = E\left\{d^2(t)\right\} &= E\left\{\left[f(y) - \hat{f} \cdot y(t)\right]^2\right\} \\ &= E\left\{[f(y)]^2 - 2\hat{f} \cdot y(t)[f(y)] + \hat{f}^2 \cdot y^2(t)\right\} \end{aligned} \tag{32}$$

To minimize T, Eq. 33 must be true.

$$\frac{\partial T}{\partial \hat{f}} = 0 = E\left\{[0] - 2y(t)f(y) + 2[\hat{f} \cdot y^2(t)]\right\} \tag{33}$$

Thus,

$$\begin{aligned} \hat{f} &= E\left\{y(t)f(y)\right\} \left[E\left\{y^2(t)\right\}\right]^{-1} \\ &= \left[\int_{-\infty}^{\infty} y(t)f(y)p(y)dy\right]\left[\int_{-\infty}^{\infty} y^2(t)p(y)dy\right]^{-1} \end{aligned} \tag{34}$$

Since y(t) is assumed to be a Gaussian random variable,

$$\begin{aligned} p(y) &= \text{probability density function of } y \\ &= \frac{1}{\sqrt{2\pi}} \frac{1}{\sqrt{\sigma}} e^{-(y-\bar{y})^2/2\sigma^2} \end{aligned} \tag{36}$$

where σ = the standard deviation and $\bar{y}$ = the mean. Substituting p(y) into Eq. 34 and assuming y(t) is a zero-mean process, $\hat{f}$ becomes

$$f = \frac{2}{\pi} \int_{-\infty}^{-\frac{\gamma}{\sigma\sqrt{2}}} e^{-w^2} dw = 1-\mathrm{erf}\left(\frac{\gamma}{\sigma\sqrt{2}}\right) = \mathrm{erfc}\left(\frac{\gamma}{\sigma\sqrt{2}}\right) \tag{36}$$

where $\mathrm{erfc}(\frac{\gamma}{\sigma\sqrt{2}})$ is the complementary error function of $(\frac{\gamma}{\sigma\sqrt{2}})$. Eq. 30 becomes

$$y_{p_1}(t) \simeq y_{p_2}(t) = \hat{f} \cdot y(t) + v_y(t) \tag{37}$$

Since the pilot will mentally correct for this threshold effect, the measure he uses is

$$y_{p_3}(t) \equiv \frac{y_{p_2}(t)}{\hat{f}} = y(t) + \frac{v_y(t)}{\hat{f}} \tag{38}$$

Since $0<\hat{f}\leq 1$, this has the effect of increasing the noise present in the measurement by a factor $\hat{f}^{-1}$. Therefore, to correct for this, Eq. 28 becomes

$$V_y = .01\pi\, E\left\{[\underline{y}(t)-\bar{\underline{y}}(t)][\underline{y}(t)-\bar{\underline{y}}(t)]^T\right\} \cdot \hat{f}^{-2} \tag{39}$$

Problems arise in calculating the threshold effects if the chosen visual threshold level γ is much larger than σ_y. As can be seen in Fig. 9, $\hat{f}$ is represented by the shaded area under the Gaussian distribution curve, (the total area under the curve is equal to 1.0). If $\gamma \gg \sigma_y$, then $\hat{f}$ becomes very small, and $\hat{f}^{-1}$ gets very large. Figure 10 shows the relationship between $\hat{f}$, γ, and σ_y. If the value of γ is such that $\sigma_y/(\hat{f} \cdot \gamma) \rightarrow \infty$,

then the solution to the iterative V_y determination process will not converge. Hence, the proper calculation of the visual threshold value is critical.

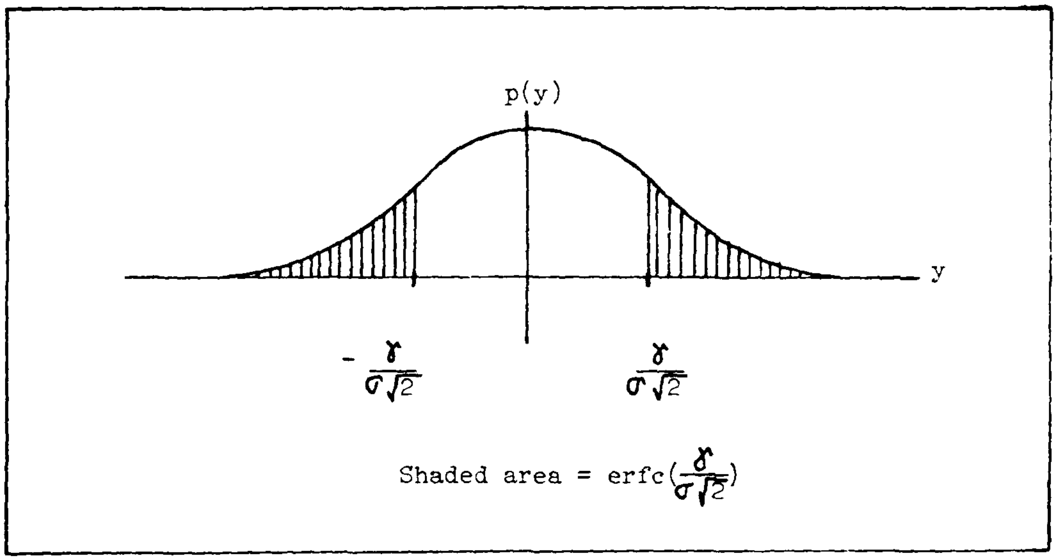


Figure 9 Complimentary Error Function

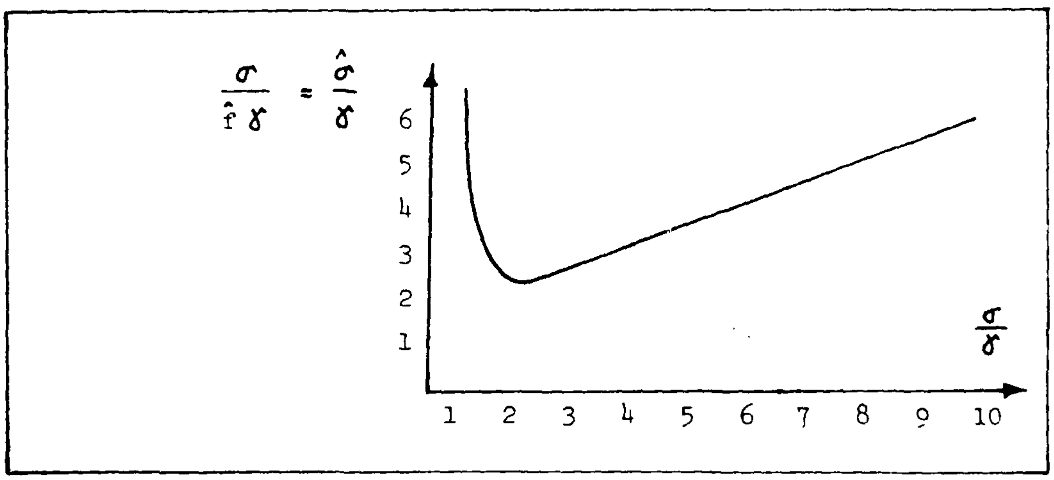


Figure 10 The Effects of $\gamma >> \sigma$

Summary. At this point, the portion of the optimal pilot model that deals strictly with the human pilot has been developed from the

optimal control system. Human response characteristics, (state estimation, internal time delay, neuromuscular lag, visual threshold, remnant), and the effects of random external disturbances have been modeled. The proper methods of determining the optimal feedback gain, observation and motor noise matrices have been discussed. What remains of the optimal pilot model to be explained is the modeling of the aircraft/control system. This explanation follows in Chapter III.

III Modeling the Aircraft/Control System

Aircraft/Control System

The figure below depicts the major components of the aircraft system. The discussion of the modeling of the aircraft/control system will address each of these components.

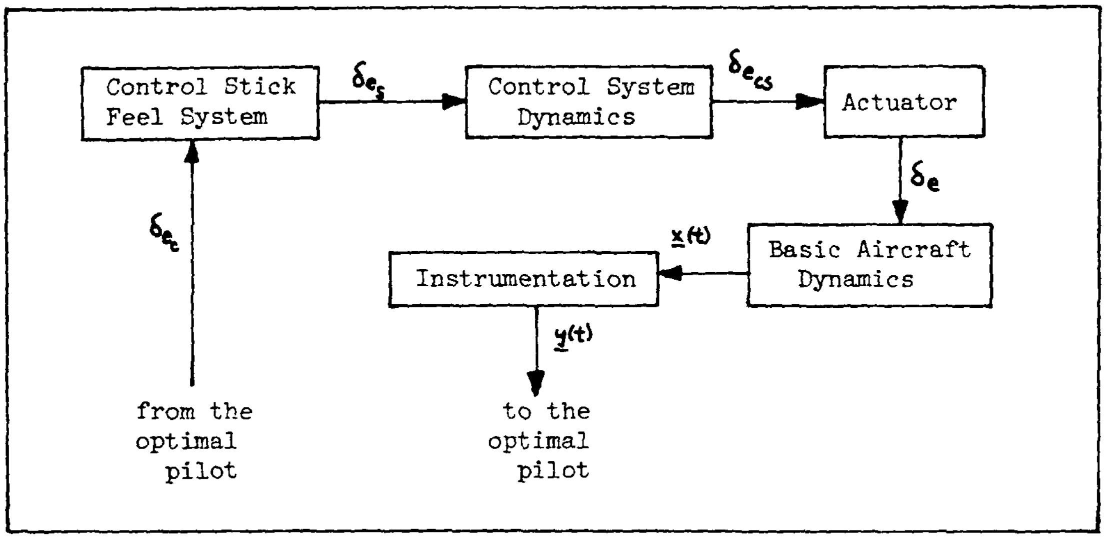


Figure 11 The Aircraft System

Basic Aircraft Dynamics. The basic aircraft equations of motion are (Ref. 1)

$$\begin{bmatrix} \Delta\dot{u} \\ \Delta\dot{w} \\ \Delta\dot{q} \\ \Delta\dot{\theta} \end{bmatrix} = \begin{bmatrix} X_u & X_w & -W_o & -g\cos\theta_o \\ Z_u & Z_w & U_o & -g\sin\theta_o \\ M_u & M_w & M_q & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \Delta u \\ \Delta w \\ \Delta q \\ \Delta\theta \end{bmatrix} + \begin{bmatrix} X_{\delta_e} \\ Z_{\delta_e} \\ M_{\delta_e} \\ 0 \end{bmatrix} \delta_e \tag{40}$$

It should be noted that the equations of motion given by Ref. 1 have been rearranged in order to imbed the $M_{\dot{w}}$ and $Z_{\dot{w}}$ terms within the remaining terms.

Thus, the stability derivatives above are actually equivalent values and include the effects of $M_{\dot{w}}$ and $Z_{\dot{w}}$. The pilot is unable to directly observe most of the variables listed above. Though the cockpit instrumentation will display some of these variables, most of the pilot information about the aircraft will consist of linear combinations of the actual states. One important item of information is the cockpit display of the error in tracking along the ILS glide slope. The error is in the form of $\Delta\epsilon$, the angle above the nominal glide path angle Γ_o. In order to relate $\Delta\epsilon$ to the dynamics of the aircraft, it is necessary to add additional states to the basic aircraft dynamic equations. These states are perturbations in altitude Δh and perturbations in slant range Δr. The states and coordinate systems are depicted in Fig. 12. From the figure the following can be seen:

$$\hat{e}_3 = (-\sin\theta)\,\hat{X}_b + (\cos\theta)\,\hat{Z}_b \tag{41}$$

$$\hat{e}_{gs} = -\cos(\Gamma_o + \theta)\,\hat{X}_b - \sin(\Gamma_o + \theta)\,\hat{Z}_b \tag{42}$$

Thus,

$$\dot{h} = U\sin\theta - W\cos\theta = \dot{h}_o + \Delta\dot{h} = f_1(U, W, \theta) \tag{43}$$

$$\begin{aligned} \dot{r} &= -U\cos(\Gamma_o + \theta) - W\sin(\Gamma_o + \theta) \\ &= \dot{r}_o + \Delta\dot{r} = f_2(U, W, \theta) \end{aligned} \tag{44}$$

The perturbation equations are

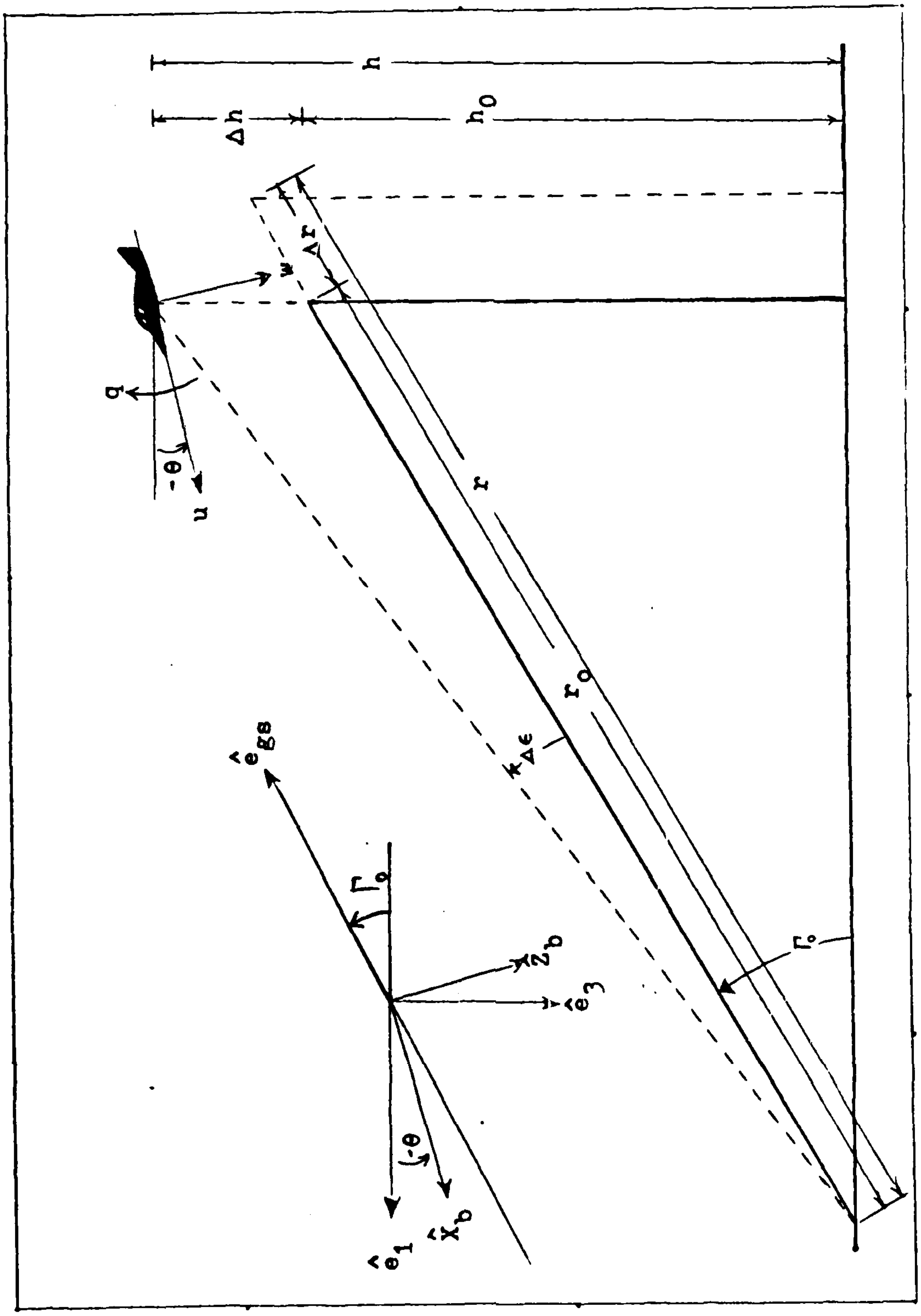


Figure 12 System States and Coordinate Systems

$$\Delta\dot{h} = \left.\frac{\partial f_1}{\partial U}\right|_o \Delta u + \left.\frac{\partial f_1}{\partial W}\right|_o \Delta w + \left.\frac{\partial f_1}{\partial \theta}\right|_o \Delta\theta$$

$$= \Delta u \sin \theta_o - \Delta w \cos \theta_o$$

$$+ (U_o \cos \theta_o + W_o \sin \theta_o)\Delta\theta \qquad (45)$$

$$\Delta\dot{r} = \left.\frac{\partial f_2}{\partial U}\right|_o \Delta u + \left.\frac{\partial f_2}{\partial W}\right|_o \Delta w + \left.\frac{\partial f_2}{\partial \theta}\right|_o \Delta\theta$$

$$= -\Delta u \cos (\Gamma_o + \theta_o) - \Delta w \sin (\Gamma_o + \theta_o)$$

$$+ \Delta\theta[U_o \sin (\Gamma_o + \theta_o) - W_o \cos (\Gamma_o + \theta_o)] \qquad (46)$$

It is reasonable to assume that the nose of the aircraft is pointed approximately along the glide slope; thus,

$$\theta_o = -\Gamma_o \qquad (47)$$

Equations 45 and 46 become

$$\Delta\dot{h} = -\Delta u \sin \Gamma_o - \Delta w \cos \Gamma_o$$

$$+ (U_o \cos \Gamma_o - W_o \sin \Gamma_o)\Delta\theta \qquad (48)$$

$$\Delta\dot{r} = -\Delta u - W_o\Delta\theta \qquad (49)$$

The nominal glide path angle used was .045 rads (2.5°); thus, Γ_o was

assumed to be a small angle. Equations 48 and 49 become

$$\Delta\dot{h} = -\Delta u\Gamma_o - \Delta w + (U_o - W_o\Gamma_o)\Delta\theta \tag{50}$$

$$\Delta\dot{r} = -\Delta u - W_o\Delta\theta \tag{51}$$

The basic aircraft dynamics become

$$\begin{bmatrix}\Delta\dot{u}\\ \Delta\dot{w}\\ \Delta\dot{q}\\ \Delta\dot{\theta}\\ \Delta\dot{h}\\ \Delta\dot{r}\end{bmatrix} = \begin{bmatrix} X_u & X_w & -W_o & -g & 0 & 0\\ Z_u & Z_w & U_o & g\Gamma_o & 0 & 0\\ M_u & M_w & M_q & 0 & 0 & 0\\ 0 & 0 & 1 & 0 & 0 & 0\\ \Gamma_o & -1 & 0 & (U_o - W_o\Gamma_o) & 0 & 0\\ -1 & 0 & 0 & -W_o & 0 & 0\end{bmatrix}\begin{bmatrix}\Delta u\\ \Delta w\\ \Delta q\\ \Delta\theta\\ \Delta h\\ \Delta r\end{bmatrix} + \begin{bmatrix}X_{\delta e_{cs}}\\ Z_{\delta e_{cs}}\\ M_{\delta e_{cs}}\\ 0\\ 0\\ 0\end{bmatrix}\delta e_{cs} \tag{52}$$

<u>Stick Feel System/Actuator</u>. The dynamics for the control stick feel system and the actuator for the NT-33A are

$$\text{Feel system: } \frac{\delta e_s}{\delta e_c} = \frac{84.5}{s^2 + 36.4s + 676}$$

$$\text{Actuator: } \frac{\delta e}{\delta e_{cs}} = \frac{5625}{s^2 + 105s + 5625}$$

These systems were dropped from the model because of their high response frequencies. It was assumed that for the human controller problem the frequencies of interest would be in the 0 to 16 rad/sec range.

Control System. Various control systems (listed in Appendix D) were tested in this thesis. To demonstrate how the basic aircraft dynamics were augmented by these control systems, a typical configuration is chosen as an example. Control System 7 is described by

$$\frac{\delta_{e_{cs}}}{\delta_{e_c}} = \frac{144}{s^2 + 16.8s + 144}$$

In phase variable form this becomes

$$x_1 = \delta_{e_{cs}} \; ; \; \dot{x}_1 = x_2$$

$$x_2 = \dot{\delta}_{e_{cs}} \; ; \; \dot{x}_2 = -16.8\, x_2 - 144\, x_1 + 144\, \delta_{e_c}$$

The aircraft dynamics, including the effects of external disturbances and motor noise as indicated by Eq. 24, are now described by Eq. 53 on the next page.

An additional control input, thrust control δ_{TH_c}, was added because the elevator alone will not adequately control the aircraft on the glide slope. This throttle term comes from the realization that the thrust forces acting on the aircraft are

$$F_{Th} = Th \cos \nu\, \hat{X}_b + Th \sin \nu\, \hat{Y}_b$$

where ν is the angle between the engine thrust axis and the aircraft centerline. Assuming that $\nu = 0$ and that there are no lateral forces due to thrust, the forces acting on the aircraft due to thrust are

$$\begin{bmatrix} \Delta\dot{u} \\ \Delta\dot{w} \\ \Delta\dot{q} \\ \Delta\dot{\theta} \\ \Delta\dot{h} \\ \Delta\dot{r} \\ \Delta\dot{\delta}_{e_{cs}} \\ \Delta\dot{x}_2 \end{bmatrix} = \begin{bmatrix} X_u & X_w & -W_o & -g & 0 & 0 & X_{\delta_{e_{cs}}} & 0 \\ Z_u & Z_w & U_o & g\Gamma_o & 0 & 0 & Z_{\delta_{e_{cs}}} & 0 \\ M_u & M_w & M_q & 0 & 0 & 0 & M_{\delta_{e_{cs}}} & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ -\Gamma_o & -1 & 0 & (U_o - W_o\Gamma_o) & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & -W_o & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & -144 & -16.8 \end{bmatrix} \begin{bmatrix} \Delta u \\ \Delta w \\ \Delta q \\ \Delta\theta \\ \Delta h \\ \Delta r \\ \Delta\delta_{e_{cs}} \\ \Delta x_2 \end{bmatrix} + \begin{bmatrix} 0 & .4 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 144 & 0 \end{bmatrix} \begin{bmatrix} \delta_{e_c} \\ \delta_{Th_c} \end{bmatrix} + \begin{bmatrix} -X_w & 0 \\ -Z_w & 0 \\ -M_w & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} w_g(t) \\ \dfrac{V_u(t)}{\tau_n} \end{bmatrix} \tag{53}$$

$$\dot{\tilde{x}}(t) = A_1\tilde{x}(t) + B_1 u^*(t) + D_2 w_{g_1}(t)$$

$$F_{Th} = Th\ \hat{X}_b$$

Adding the perturbation thrust to the Δu equation of motion yields

$$\Delta\dot{u} = X_u \Delta u + X_w \Delta w - W_o \Delta q + X_{\delta_{e_{cs}}} \Delta\delta_{e_{cs}} + \frac{\Delta Th}{m_1} - X_w w_g \qquad (54)$$

For an ideal jet, ΔTh is dependent only on altitude and power setting. For the small altitude changes examined in this thesis, ΔTh is dependent only on the power setting δ_{Th_c}. Thus,

$$\frac{\Delta Th}{m_1} = \left[\frac{Th_{max}}{m_1}\right] \delta_{Th_c} \qquad (5\_)$$

where Th_{max} is the maximum thrust available and m_1 is the weight of the aircraft. For the NT-33A, Th_{max} is approximately 5,000 lb_f and the weight for the approach is approximately 12,500 lb_m. Thus, the coefficient for the δ_{Th_c} term is

$$\frac{\Delta Th}{m_1} = 0.4$$

For this thesis, the state variable Δr was not included in the modeling of the aircraft/control system. The optimal pilot computer program subroutine designed to solve Eq. 7 (subroutine MRIC in Appendix C) would not converge towards a solution if this state was included in the model. No reason for this difficulty could be determined, so the Δr state was dropped and the value of r for all applicable terms in the

displayed variable coefficient matric C (to be discussed next) was approximated as r_o, the nominal slant range.

Cockpit Instrumentation. As previously mentioned, the pilot is unable to observe most of the aircraft states directly. The cockpit instrumentation displays some of the states directly, but most of the information displayed is a linear combination of the states. The instrumentation available to the NT-33A pilot for the glide slope task is depicted in Fig. 13. It is usually assumed that if the pilot can read a displayed quantity explictly, then he can implictly obtain information about the rate-of-change of that quantity. However, instrument lag due to hysteresis causes doubt concerning the usefulness of implicit rate information during an ILS landing approach. Therefore, the displayed variables modeled in this thesis were

(1) Perturbation angle of attack $(\Delta\alpha)$

(2) Perturbation rate-of-change of angle of attack $(\Delta\dot{\alpha})$

(3) Perturbation pitch attitude $(\Delta\theta)$

(4) Perturbation rate-of-change of pitch attitude $(\Delta\dot{\theta})$

(5) Perturbation airspeed (Δu)

(6) Perturbation descent rate $(\Delta\dot{h})$

(7) Perturbation angle above the nominal glide slope angle $(\Delta\epsilon)$

Those displayed variables that are linear combinations of the system states are developed as follows:

$$\Delta\alpha = \frac{1}{U_o}\Delta w \tag{56}$$

$$\Delta\dot{\alpha} = \frac{1}{U_o}\Delta\dot{w} = \frac{Z_u}{U_o}\Delta u + \frac{Z_w}{U_o}\Delta w + \Delta q + \frac{Z_{\delta_{e_{cs}}}}{U_o}\Delta\delta_{e_{cs}} \tag{57}$$

$$\Delta\dot{h} = -\Gamma_o\Delta u - \Delta w + (U_o - W_o\Gamma_o)\Delta\theta \tag{58}$$

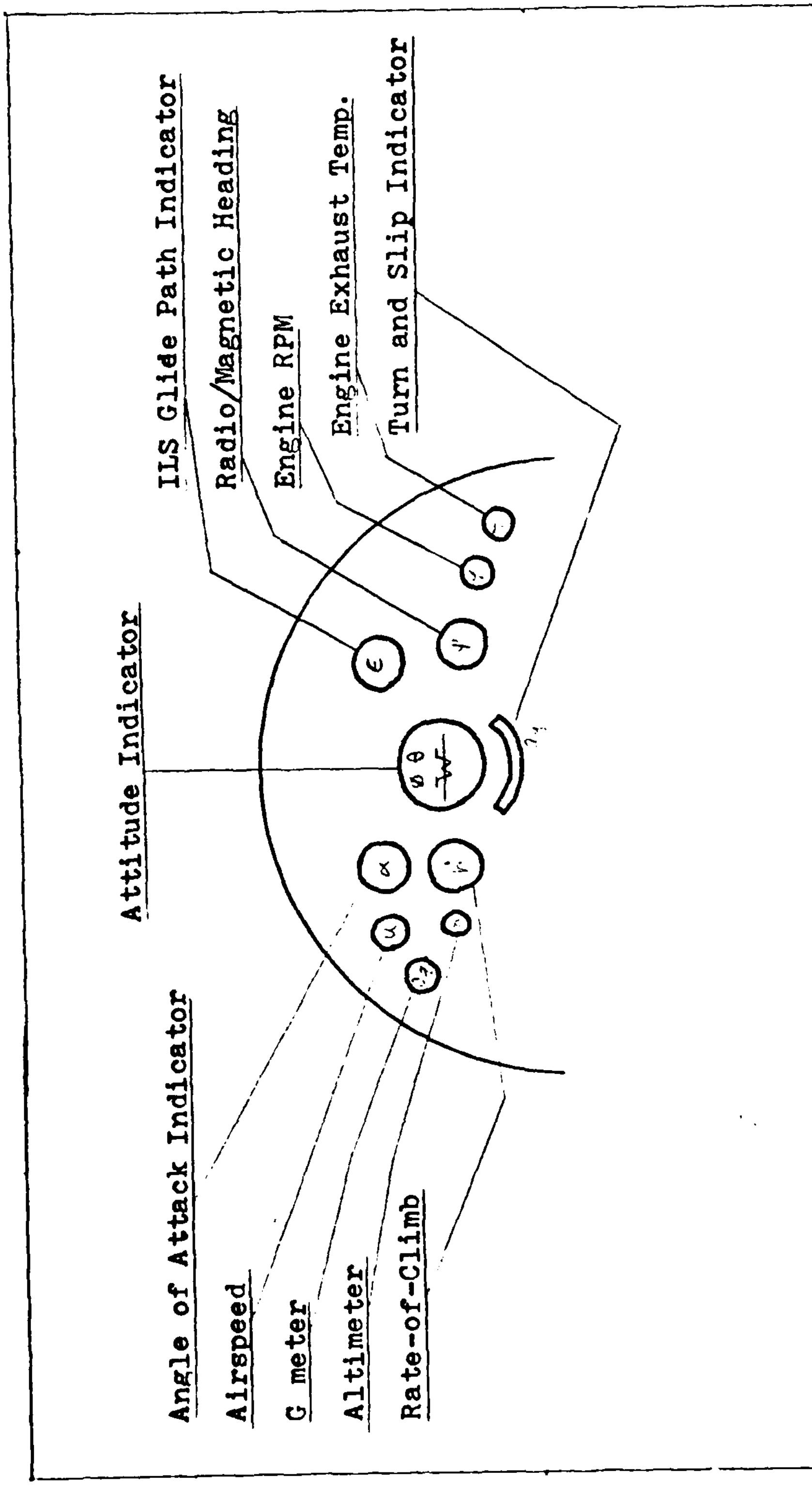


Figure 13 NT-33A Cockpit Instrumentation

The display variable $\Delta\epsilon$ is developed as follows:

$$\epsilon = \epsilon_o + \Delta\epsilon \; ; \; \epsilon_o = 0$$

$$\epsilon_o + \Delta\epsilon \approx \frac{h-h_o}{r} = \frac{h - r \sin \Gamma_o}{r}$$

$$= \left[\frac{h_o}{r_o} - \sin \Gamma_o\right] + \left[\frac{\partial\left(\frac{h}{r} - \sin \Gamma_o\right)}{\partial h}\Bigg|_o\right] \Delta h$$

$$\left[\frac{\partial\left(\frac{h}{r} - \sin \Gamma_o\right)}{\partial r}\Bigg|_o\right] \Delta r + \text{H.O.T.s}$$

$$\Delta\epsilon = \frac{\Delta h}{r_o} - \frac{h_o}{r_o^2} \Delta r \qquad (59)$$

The displayed variables are now determined according to Eq. 60 on the next page.

Weighting Matrices. Determination of the values of the weighting matrices is a difficult task. While the aircraft dynamics can be calculated from experimental and analytical data, the calculation of the weighting matrices is more subjective. Hess suggested that, in order for an optimal pilot model to predict the pilot's rating, the elements of the weighting matrices should be the reciprocal squares of the maximum desirable perturbations of their corresponding variables (Ref. 7). The change in any variable would be viewed as a fraction of the maximum desirable perturbation. The penalty assessed by the performance index for a fractional change in one variable would be equal to the penalty for the same fractional change in another variable. For example, consider two input variables;

$$\begin{bmatrix} \Delta\alpha \\ \Delta\dot{\alpha} \\ \Delta\theta \\ \Delta\dot{\theta} \\ \Delta\dot{h} \\ \Delta u \\ \Delta\epsilon \end{bmatrix} = \begin{bmatrix} 0 & \frac{1}{U_o} & 0 & 0 & 0 & 0 & 0 & 0 \\ \frac{Z_u}{U_o} & \frac{Z_w}{U_o} & 1 & 0 & 0 & 0 & Z_{\delta e_{cs}} & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ -\Gamma_o & -1 & 0 & (U_o - W_o\Gamma_o) & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{r_o} & 0 & -\frac{h_o}{r_o^2} & 0 \end{bmatrix} \begin{bmatrix} \Delta u \\ \Delta w \\ \Delta q \\ \Delta\theta \\ \Delta h \\ \Delta\delta_{e_{cs}} \\ \Delta x_2 \end{bmatrix} \qquad (60)$$

$$\underline{y}(t) = C\,\underline{x}(t)$$

elevator perturbations $\Delta\delta_e$ and thrust perturbations $\Delta\delta_{Th}$. Figure 14 shows that, if the weightings are chosen as Hess suggested, a change of 10% of the maximum desirable perturbation of $\Delta\delta_e$ would incur the same penalty as a 10% change in $\Delta\delta_{Th}$. The optimal controller, in attempting to minimize the penalties assessed against all variables, will not favor one variable over another, (though it would if the penalty for a fractional change in one variable were more than for the same fractional change in another variable).

Difficulty arises in determining the value of the maximum desirable perturbation. Obviously, these values will differ from pilot to pilot. This means that a choice of weighting values would yield an optimal pilot model that would predict the ratings of only the pilot whose subjective weighting values were used. It was assumed, however, that well-trained, skillful pilots do not differ significantly in what they consider acceptable control and state perturbations during an approach, and that these differences would have very little effect on the results of the optimal pilot model.

In choosing the weightings for this thesis, the opinions of the CALSPAN pilots who flew the actual missions were not available, so the opinions of available pilots experienced in fighter aircraft were used. These weightings are depicted in Fig. 15.

It was felt that the pilot cannot make reasonably accurate assessments of the maximum desirable perturbations of Δq and $\Delta x_2 = \Delta\dot{\delta}_e$, so the weightings were set to zero. This would have the effect of telling the optimal controller that perturbations in these quantities are not important to the pilot. Actually, by weighting other variables, these variables will remain well-behaved. The weighting on $\Delta\delta_{e_c}$ was originally

	$\Delta\delta_e$	$\Delta\delta_{Th}$
Maximum Desirable Perturbation Q_i	5° = .087 rads	$.10 \frac{Th}{Th_{max}}$
Weighting Element $q_i = \frac{1}{Q_i}$	131.3	100
10% change in variable	.0087 rads	$.01 \frac{Th}{Th_{max}}$
Penalty Assessed by Performance Index $J_i = q_i^{\,2}$	J_i = .01	J_i = .01

Figure 14 Weighting Matrix Example

Variable	Q_i	q_i
Δu	2 kts = 3.338 ft/sec	0.088
Δw	3 kts = 5.07 ft/sec	0.039
Δq	———	0.0
$\Delta\theta$	5° = 0.087 rads	131.3
Δh	5.0 ft	0.04
$\Delta\delta e_c$	———	0.0
$\Delta\delta Th_c$	$0.5 \frac{Th}{Th_{max}}$	400.0
$\Delta\delta_e$	5° = 0.087 rads	0.039
Δx_2	———	0.0

Figure 15 Weighting Values

set to 131.3 (5° perturbation). This was later set to zero to allow the desired value for the neuromuscular lag to be achieved. This is discussed further in Chapter IV.

Summary. The aircraft/control system modeling has been explained and the developement of the optimal pilot model is now complete. Chapter IV will now discuss the organization and construction of the optimal pilot computer program, OPSACT.

IV The Optimal Pilot Computer Program

The optimal pilot computer program, OPSACT, (Optimal Pilot - Single Axis Control Task), is comprised of four parts. The first part consists of subroutines for the inputting of variables and the calculation of constants. The second part handles the calculation of the optimal feedback gain matrix L_1 and the determination of the neuromuscular lag τ_n. The third part determines the input and displayed variable variances and calculates the estimator gain matrix S. The fourth part is the pilot/aircraft system integration loop.

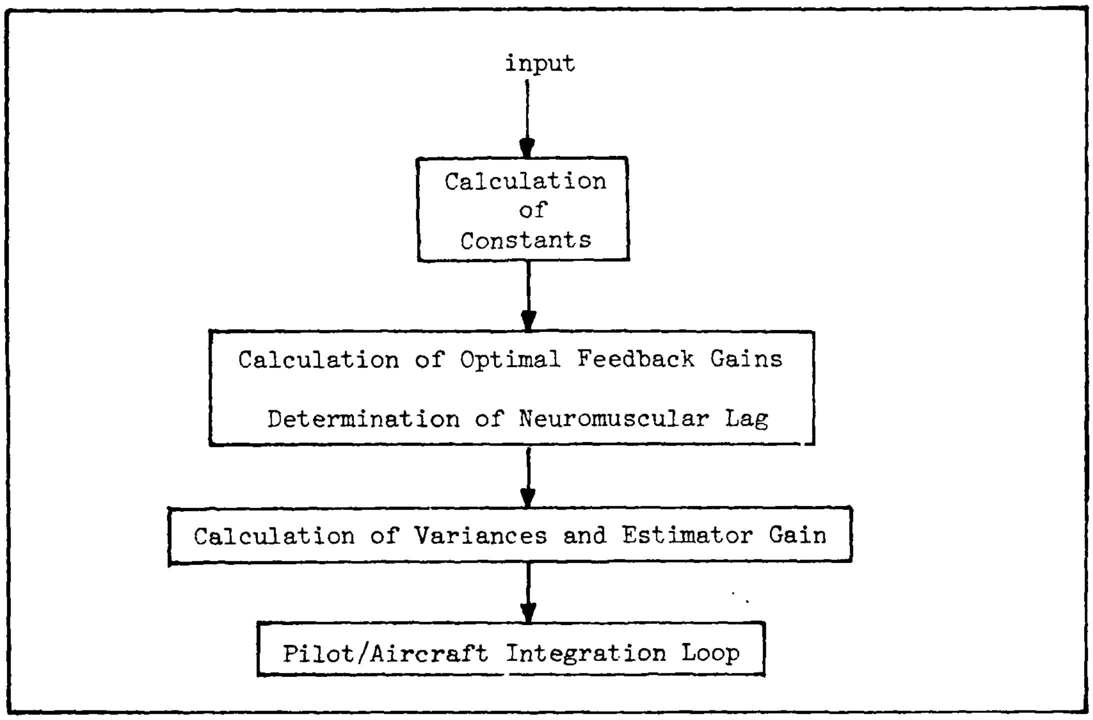


Figure 16 The Major Divisions of OPSACT

Inputs/Constants

The input variables to program OPSACT are listed in the following figure.

A_o	Aircraft/Control/Equivalent Lag (A/C/EL) System Coefficient Matrix.
B_o	A/C/EL System Input Coefficient Matrix.
C	Displayed Variables (Instrumentation) Matrix.
Q_o	A/C/EL System State Weighting Matrix.
R_o	A/C/EL System Input Weighting Matrix.
V_y	Displayed Variable (Observation Noise) Covariance Matrix
W_1	External Disturbance/Motor Noise Covariance Matrix
t_o	Initial Time of Task
t_f	Final Time of Task
Δt	Integration Time Increment
x_o	Initial State Vector
u_o	Initial Input Vector
γ	Visual Tresholds

Figure 17 OPSACT Inputs

To solve Eq. 10, OPSACT integrates from t_o to t_f using transition matrices. Eq. 10 becomes

$$\underline{\tilde{x}}[(k+1)\Delta t] = e^{A_1 \Delta t}\underline{\tilde{x}}(k\Delta t) + \int_{k\Delta t}^{(k+1)\Delta t} e^{A_1[(k+1)\Delta t-\sigma]} B_1 \underline{u}(\sigma)d\sigma$$

$$= \phi(\Delta t)\underline{\tilde{x}}(k\Delta t) + \Gamma(\Delta t)\underline{u}(k\Delta t) \tag{61}$$

The estimator equation, (Eq. B-6, Appendix B), becomes

$$\hat{\tilde{\underline{x}}}[(k+1)\Delta t] = e^{A_1 \Delta t}\hat{\tilde{\underline{x}}}(k\Delta t) + \int_{k\Delta t}^{(k+1)\Delta t} e^{A_1[(k+1)\Delta t-\sigma]} B_1 \underline{u}(\sigma) d\sigma$$

$$+ \int_{k\Delta t}^{(k+1)\Delta t} e^{A_1[(k+1)\Delta t-\sigma]} S[\underline{y}(k\Delta t)-C\hat{\tilde{\underline{x}}}(k\Delta t)]d\sigma$$

$$= \phi(\Delta t)\hat{\tilde{\underline{x}}}(k\Delta t) + \Gamma(\Delta t)\underline{u}(k\Delta t) + \Psi(\Delta t)[\underline{y}(k\Delta t)-C\hat{\tilde{\underline{x}}}(k\Delta t)] \tag{62}$$

The predictor equation, (Eq. B-20), becomes

$$\hat{\tilde{\underline{x}}}^*[(k+1)\Delta t] = e^{A_1 \Delta t}\hat{\tilde{\underline{x}}}^*(k\Delta t) + \int_{k\Delta t}^{(k+1)\Delta t} e^{A_1[(k+1)\Delta t-\sigma]} B_1 \underline{u}^*(\sigma) d\sigma$$

$$= \phi(\Delta t)\hat{\tilde{\underline{x}}}^*(k\Delta t) + \Gamma(\Delta t)\underline{u}^*(k\Delta t) \tag{63}$$

Since Δt = constant for this thesis, $\phi(\Delta t) = e^{A_1 \Delta t}$ = constant. By making the transformation $\xi = [(k+1)\Delta t-\sigma]$, then $\Gamma(\Delta t)$ and $\Psi(\Delta t)$ become

$$\Gamma(\Delta t) = \int_{k\Delta t}^{(k+1)\Delta t} e^{A_1[(k+1)\Delta t-\sigma]} B_1 d\sigma = \int_{o}^{\Delta t} e^{A_1 \xi} d\xi B_1$$

$$= \text{constant} \tag{64}$$

$$\Psi(\Delta t) = \int_{k\Delta t}^{(k+1)\Delta t} e^{A_1[(k+1)\Delta t-\sigma]} Sd = \int_{o}^{\Delta t} e^{A_1 \xi} d\xi S$$

$$= \text{constant} \tag{65}$$

Thus, ϕ, Γ, and another constant used by the predictor, $e^{A_1 \tau}$, are calculated outside of the pilot/aircraft integration loop. The estimator

gain matrix S is needed from part three of the program before the calculation of Ψ is complete.

Optimal Feedback Gain/Neuromuscular Lag

The optimal feedback gains are determined according to Eqs. 17 and 18. The iterative process of arriving at a desired value for the neuromuscular lag τ_n through repeated guesses of R and R_o, as discussed in Chapter II, can be accomplished by reentering OPSACT with different values for R and R_o. This method, however, consumes a great deal of computer time. At present, there is no feature in OPSACT to allow the entering of desired values of τ_n and letting the computer vary R and R_o in order to obtain the desired lag. For this thesis, a second program, OPTCON, (which is designed to solve only the optimal feedback gain problem), was used to obtain the correct values of R and R_o that would yield the desired neuromuscular lag for each aircraft/control system configuration. The inputs to the optimal feedback gain problem are A_o, B_o, Q_o and R_o. No problems were encountered in arriving at the desired thrust neuromuscular lag of 1.0 second, (this lag actually included the engine lag along with the pilot lag). Considerable difficulty arose in finding values of R and R_o that would yield the desired elevator lag of 0.1 seconds. Results showed that as the corresponding element in R decreased, or the corresponding element in R_o increased, the calculated lag would increase. Initial guesses of R and R_o resulted in too small a value for τ_{n_e}. Repeated iterations resulted in the R element being reduced to zero and the R_o element being increased to the point that OPTCON could not arrive at a solution, yet the value of τ_{n_e} was still too small. Experimenting with the off-diagonal elements of R_o revealed that any non-zero entry in these elements would only decrease the value of τ_{n_e} and $\tau_{n_{Th}}$. It was found, however, that by

increasing the weighting on the commanded throttle input δ_{Th_c} in the R submatrix from 25.0, (which corresponded to a maximum desirable perturbation of 20% $\frac{Th}{Th_{max}}$), to a weighting of 400.0, (5% $\frac{Th}{Th_{max}}$), and then reguessing R_o to obtain the 1.0 second thrust lag, the δ_{e_c} element of R_o could now be increased to achieve the desired elevator lag without having computational problems. This relationship between the weighting of one input and the arrival at the desired neuromuscular lag of another input was not mentioned in any of the research material.

The final action taken by the second part of OPSACT is to take the calculated values of τ_n and form the actual system state coefficient matrix A_1 as described by Eq. 10. The values of τ_n are then stored for use in forming the external disturbance/motor noise covariance matrix W_1.

<u>Estimator Gains/Variances</u>

The estimator gain matrix S is calculated using the A_1 matrix. The third part of OPSACT forms the W_1 matrix and determines the estimator gains according to Eqs. B-17 and B-18. As explained in Chapter II, the motor noise V_{u_i} is proportional to the variance of the n+i states $u_i(t)$; i = 1,2,...m; and the observation noise V_{y_i} is proportional to the variance of the displayed variables $y_i(t)$; i = 1,2,...ny, where ny = the number of variables displayed on the instrument panel. The covariance matrix of the system states is calculated using Eq. 27. The first two terms of that equation are calculated as indicated. Noting that the third term can be written as (Ref. 14)

$$\int_o^{\infty} e^{\bar{A}\sigma} [e^{A_1\tau} PC^T V_y^{-1} CP e^{A_1^T\tau}] e^{\bar{A}^T\sigma} d\sigma = \int_o^{\infty} e^{\bar{A}\sigma} G e^{\bar{A}^T\sigma} d\sigma \quad (66)$$

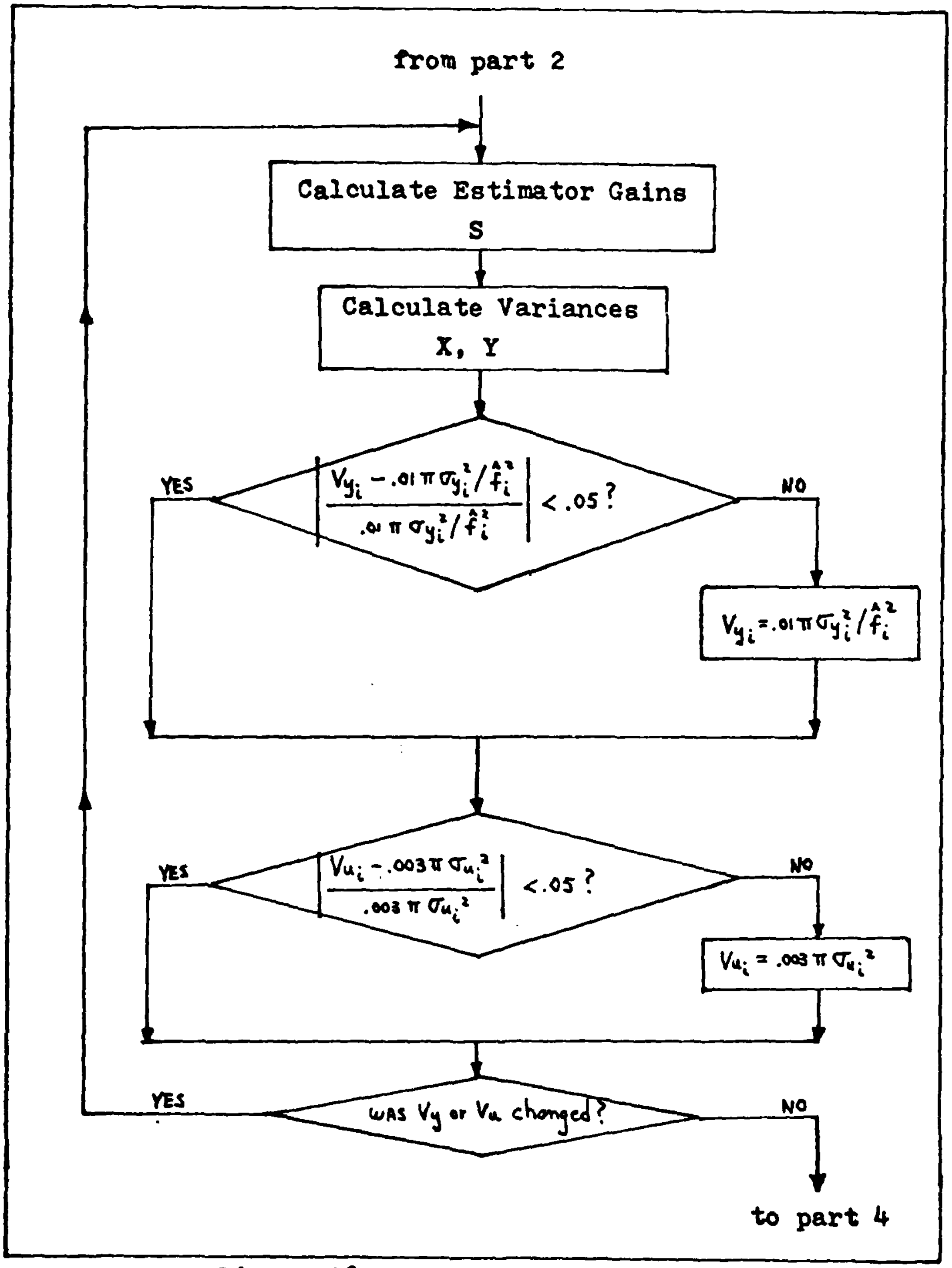


Figure 18 Part Three of OPSACT

Evaluation of the integral in Eq. 66 is equivalent to finding the solution matrix Z to the equation

$$0 = \bar{A}Z + Z\bar{A}^T + G \tag{67}$$

Eq. 67 is a Lyapunov equation and is solved to determine the third term of Eq. 27. The covariance matrix of the displayed variables is obtained from the results of Eq. 27.

$$Y = C^T XC \tag{68}$$

As depicted in Fig. 19, the n=i diagonal element of W_1; i = 1,2,...m, is multiplied by $\tau_{n_i}^2$ and is compared to $.003\pi\sigma_{u_i}^2$, where $\sigma_{u_i}^2$ is the n+i diagonal element of the state covariance matrix X. If the difference is greater than 5% of $.003\pi\sigma_{u_i}^2$, then V_{u_i} is set equal to $.003\pi\sigma_{u_i}^2$ and divided by $\tau_{n_i}^2$ to form a new W_1 matrix. Similarly, the i^{th} diagonal term of V_y; i = 1,2,...ny, is compared to the i^{th} diagonal term of Y, $.01\pi\sigma_{y_i}^2/f^2$. If the difference is greater than 5% of $.01\pi\sigma_{y_i}^2/f^2$, then V_y is set equal to $.01\pi\sigma_{y_i}^2/f^2$. If either V_u or V_y was changed, the program returns to the start of part three and repeats the process until all elements of V_u and V_y are within 5% of the desired values. The program then continues to part four.

Pilot/Aircraft Integration Loop

Part four of OPSACT combines the actions of the aircraft system, the estimator, and the predictor to form the pilot/aircraft iteration loop. This process divides the continuous actions of the pilot/aircraft system into small increments, where each increment represents one pass through the

pilot/aircraft loop. Integration is accomplished by means of the transition matrices developed in part one.

$$W_1 = \left[\begin{array}{c|c} W & 0 \\ \hline 0 & V_u/\tau_n^2 \end{array}\right]$$

$$= \left[\begin{array}{cccc|cccc} w_{1,1} & & & & & & & \\ & w_{2,2} & & & & \underline{0} & & \\ & & \ddots & & & & & \\ & & & w_{n,n} & & & & \\ \hline & & & & \left(\frac{V_u}{\tau_n^2}\right)_{n+1,n+1} & & & \\ & \underline{0} & & & & \left(\frac{V_u}{\tau_n^2}\right)_{n+2,n+2} & & \\ & & & & & & \ddots & \\ & & & & & & & \left(\frac{V_u}{\tau_n^2}\right)_{n+m,n+m} \end{array}\right]$$

$$\left(\frac{V_u}{\tau_n^2}\right)_{n+2,n+2} \times \tau^2_{n\;\,n+2}$$

$V_{u_{n+2}}$ is compared to $.003\pi\sigma^2_{u_{n+2}}$

Figure 19 Comparing Noise to Variance

It should be noted that the order in which the calculations are made is not determined by following the arrows around Fig. 20. Though the loop models the control task in discrete steps, the actual integration process made by the pilot/aircraft system is continuous. Fig. 20 only indicates where the various inputs and outputs go. The model is not a sequential system that is initiated at the aircraft/control system block with the input/output information flowing along the arrows activating other components of the model. During any one pass through the loop, inputs exist for all of the integrations, (initial values, new values, delayed values); thus, the order in which the integrations take place is not important. Figure 21 shows which components must precede others, (for example, the delaying of $\underline{x}(t)$ must follow the determination of that quantity). The grouping of the integration processes indicates that they should be viewed as taking place simultaneously.

The choice of the integration time increment Δt is not totally arbitrary. When attempting to reconstruct a signal from discrete samples, a sampling rate that is too slow, (time increment too large), will prevent accurate reconstruction of the original signal. Figure 22 shows that if a signal S_o is sampled at a low frequency f_1, the reconstructed signal S_{r_1} will not resemble the original signal. Sampling at a higher rate f_2 will yield a reconstructed signal S_{r_2} that is a better representation of the original signal. Shannon's sampling theorem states that a signal can be accurately reconstructed if the sampling increment is chosen such that

$$\Delta t = \tfrac{1}{2}\left(\frac{2\pi}{\omega_c}\right)$$

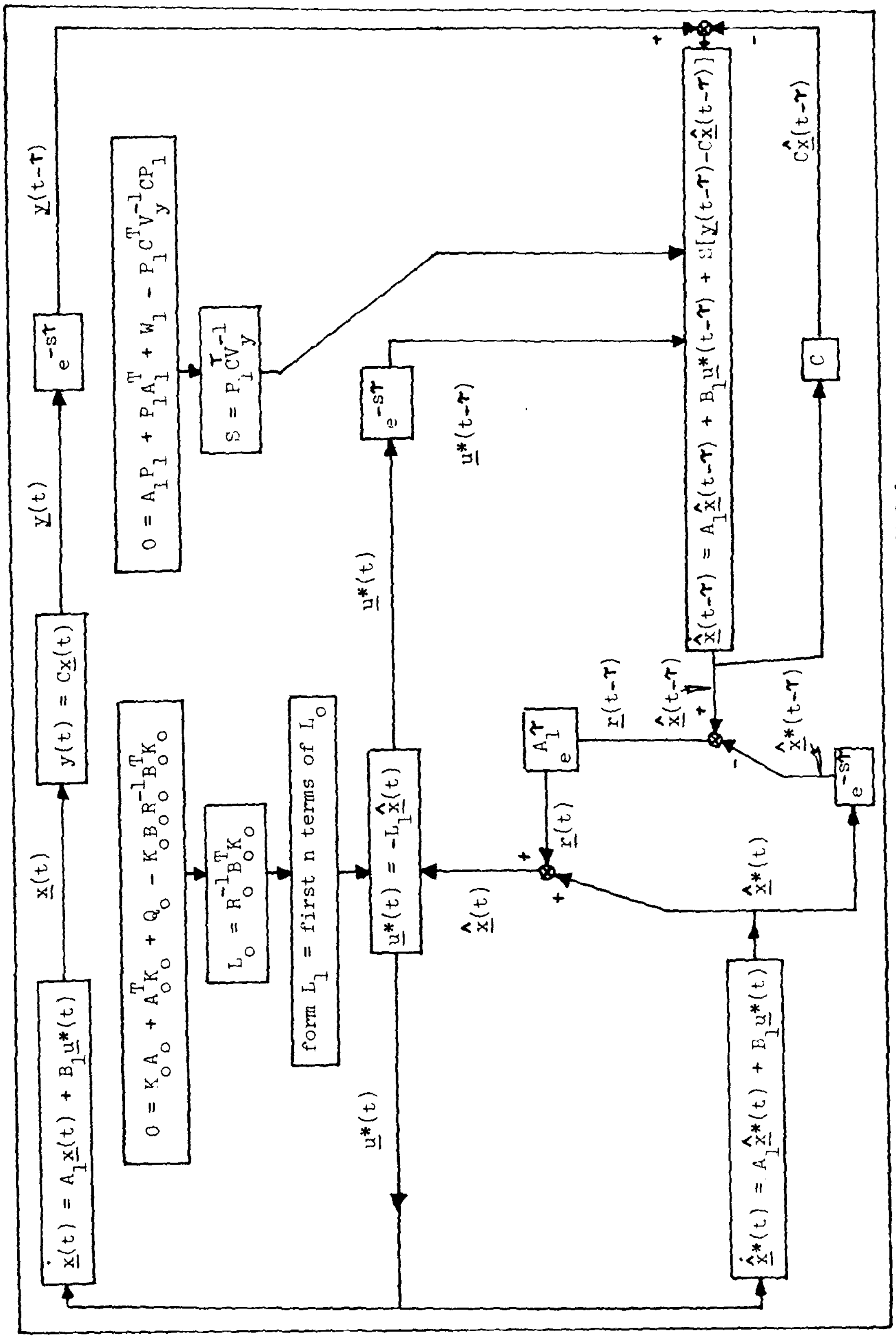


Figure 20 The Optimal Pilot Model

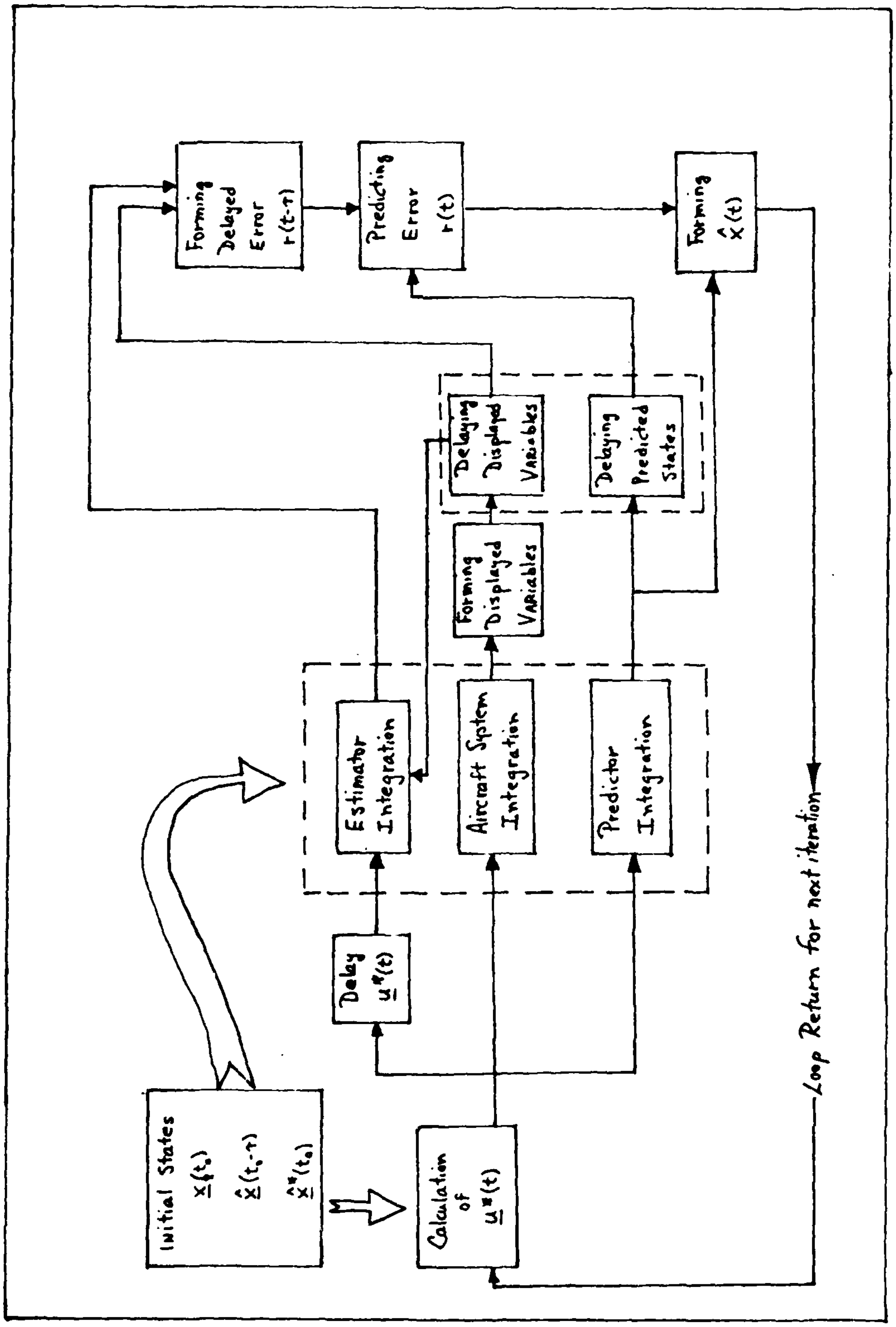


Figure 21 Order of Calculation of Components of Pilot/Aircraft Integration Loop

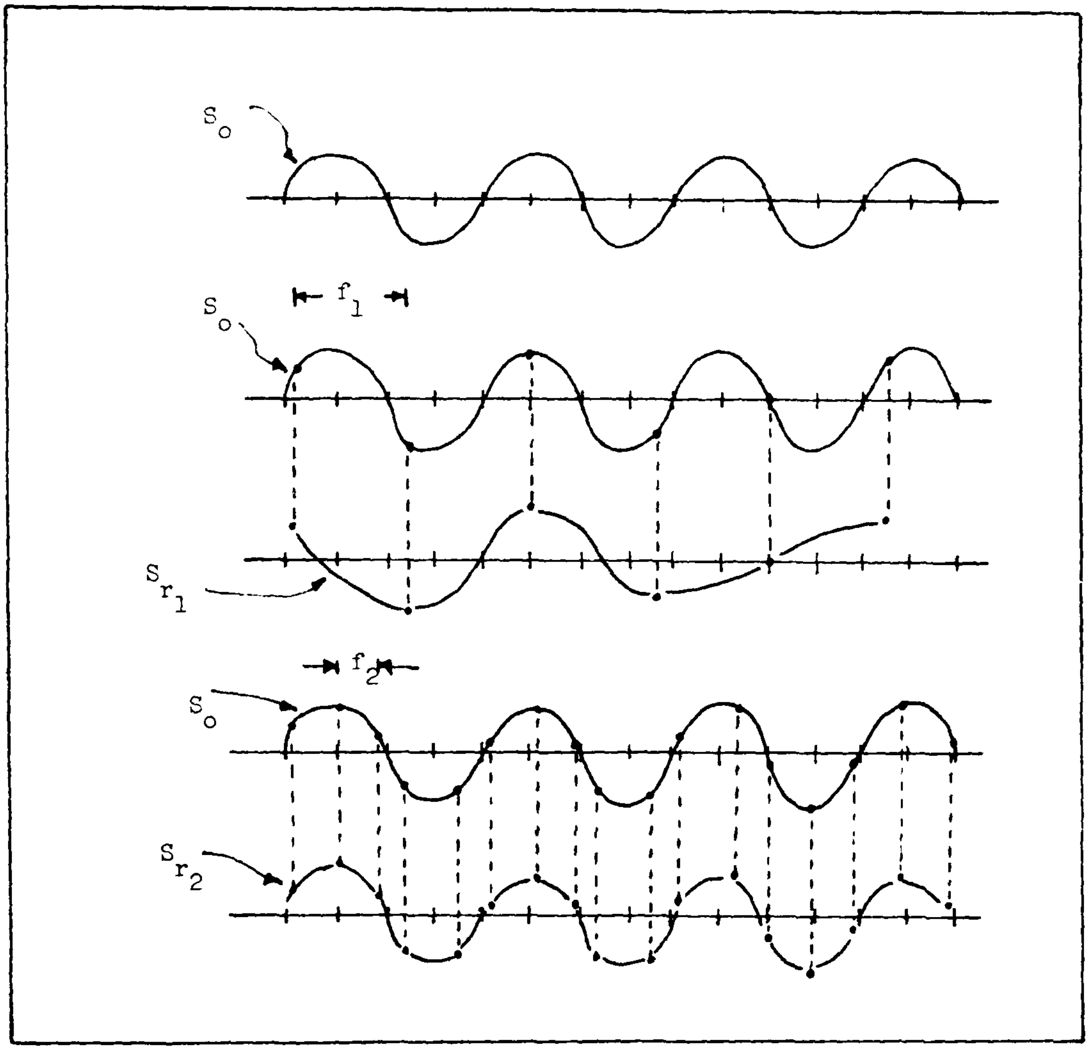


Figure 22 Signal Sampling Rate

where ω_c is the highest frequency in the signal (Ref. 12). In practice, other factors require that a higher sampling rate be used. Common choices are 6 to 20 samplings per signal cycle. For this thesis, this would mean that the integration time increment would have to be between $\frac{1}{6}\frac{2\pi}{\omega_c}$ and $\frac{1}{20}\frac{2\pi}{\omega_c}$, where ω_c is the largest eigenvalue, (highest frequency in the system response). Since the pilot's internal time delay τ is achieved by OPSACT by delaying certain quantities an integer number of loop iterations,

Δt was chosen as

$$\Delta t = \frac{1}{8}(\tau) = .025 \text{ seconds} = \frac{1}{8}(0.2) = \frac{1}{15}\left(\frac{2\pi}{\omega_c}\right) \qquad (70)$$

where ω_c was chosen as 17 rads/sec, which was the highest frequency in the response of the systems tested. This made the sampling rate approximately 15 times per second.

A listing of program OPSACT is contained in Appendix C. Included in this listing are the subroutines written by Kleinman (Ref. 15).

IV Results

In order to predict the flying qualities of aircraft, a correlation must exist between the performance index generated by the optimal pilot model and the CALSPAN pilots' opinion ratings. The results obtained from program OPSACT were compared to the CALSPAN results in an attempt to find that correlation.

CALSPAN conducted eighty-three evaluations using two flight profiles. Profile 1 consisted of an ILS approach followed by a touch-and-go landing. The pilot first evaluated the overall approach and landing, and then he rated the approach separately. Profile 2 consisted of an ILS approach only. Since the task modeled in this thesis was just the ILS approach, the overall rating given for Profile 1 flights could not be used as a basis for comparison with the optimal pilot model results.

There were forty-nine aircraft/control system configurations that were evaluated, some of which were rated using both flight profiles. CALSPAN noted that in the cases where a single configuration was evaluated using two profiles, the approach rating made on the Profile 2 flight was higher (worse) than the separate approach rating made on the Profile 1 flight (Ref. 1). This indicated that the approach rating for the latter flight profile could have been influenced to some degree by the landing instead of being based solely on the Cooper-Harper criteria. Since the model predictions were designed to be compared to the pilot opinion ratings of the ILS approach based only on Cooper-Harper criteria, Profile 1 approach ratings could not be used to compare with the model results. This left only eleven evaluations, (eight aircraft/control system configurations evaluated by two pilots), for comparison. The results of these evaluations

are summarized below.

Pilot/Flt No.	Configuration	Approach Rating
A/1883	2-1	3
A/1890	5-3	4
B/1901	5-3	3
A/1899	1-6	5
A/1889	2-4	5
B/1892	2-4	3
A/1894	4-4	5
A/1885	1-3	6
B/1898	1-3	6
A/1887	3-2	6
A/1897	3-3	7

Figure 23 CALSPAN Evaluations Used For Comparison
With Optimal Pilot Model Results

Program OPSACT simulated eight optimal pilot/aircraft configurations performing an ILS approach. Using CALSPAN nominal data, the simulation was run for 200 seconds to approximate the duration of the actual approaches. Run 1 used arbitrarily chosen initial conditions of +5 ft/sec in forward velocity and + 25 ft in altitude. The model results are listed in Fig. 24. These results are then compared to CALSPAN pilot A's opinion ratings in Fig. 25. The Cooper-Harper and performance index scales were aligned by first performing a least-squares linear regression for each plot, (that is, find the straight line that minimizes the sum of the squares of the deviations of the actual data points from the straight line of best fit). The two vertical scales were then aligned so as to

superimpose the best fit straight lines from both plots. Correlation between the pilot ratings and the model predictions would be indicated if the two plots were similar in shape and slope.

Configuration	Performance Index
2-1	7,113
5-3	10,110
1-6	8,660
2-4	11.981
4-4	12,621
1-3	8,306
3-2	14,266
3-3	13,082

Initial conditions: +5 ft/sec (airspeed)
+25 ft (altitude)

Figure 24 Optimal Pilot Model Results: Run #1

As can be seen, the plots seem to be similar except for configurations 1-6, 3-3, and especially 1-3. The mean difference between the two plots is equivalent to 1.1 Cooper-Harper rating increments. To determine whether this difference between a human pilot and the optimal pilot model is reasonable, the mean difference between the ratings of those aircraft/control system configurations evaluated by both CALSPAN pilots was examined.

Of the forty-nine configurations flown by CALSPAN, seventeen were evaluated by both pilots. The results of these evaluations are depicted in Figures 26a and 26b.

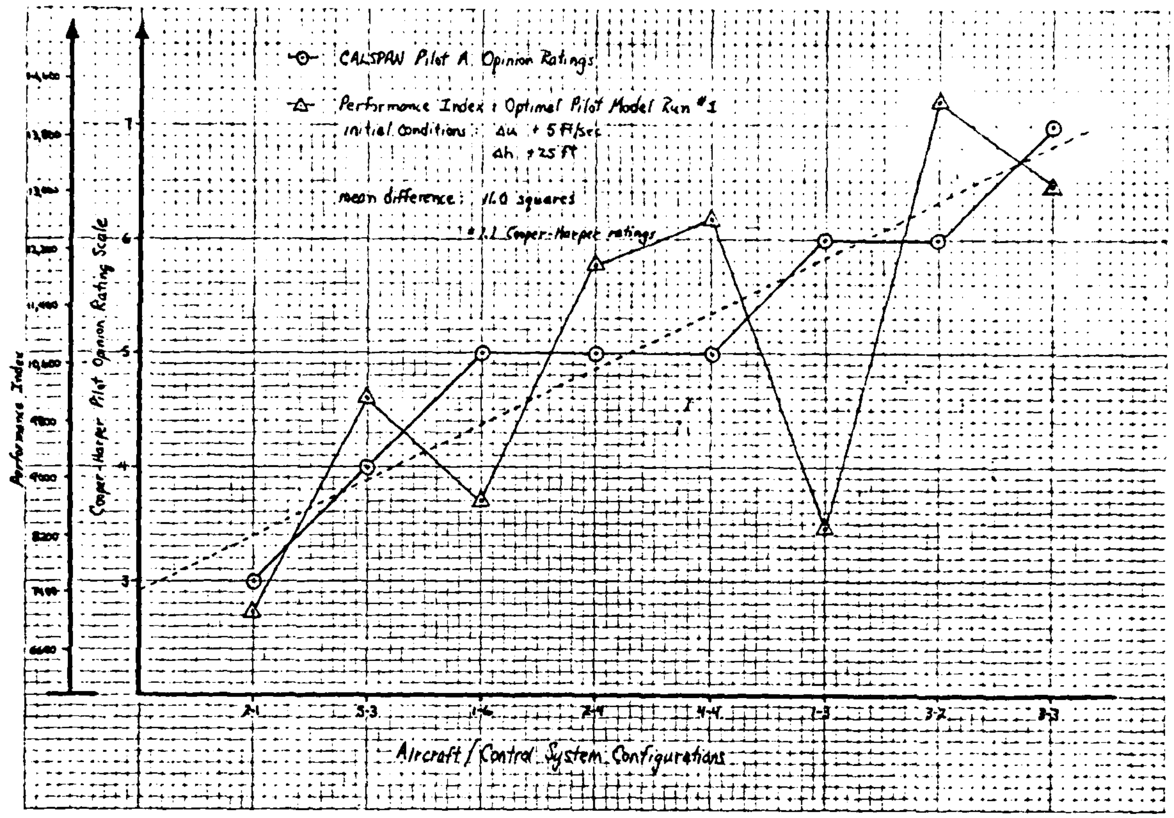


Figure 25 Comparison of Optimal Pilot Run #1 Results to CALSPAN Pilot A's Opinion Ratings

The mean difference between the opinion ratings of the two CALSPAN pilots was 1.1 Cooper-Harper rating increments. The largest difference in ratings between the optimal pilot model and CALSPAN pilot A was 2.5 rating increments, (configuration 1-3), whereas there are four configurations where the two CALSPAN pilots differed by 2.5 or more rating increments, (configurations 3-C, 3-1, 4-3, and 7-3). This indicates that significant differences between the model and the pilot are not unreasonable. The fact that the mean difference in ratings for the model-pilot A comparison and the pilot B-pilot A comparison were the same could indicate that the model is just as accurate in predicting pilot A ratings as pilot B. The limited number of

	Configuration	Pilot A Rating	Pilot B Rating
Profile #1 Overall Rating	2-1	2	2
	3-C	2	5
	4-C	3	3
	1-C	4	4
	1-1	4	4
	2-2	4.5	4
	2-A	4	6
	7-3	4	6
	3-1	4.5	7
	5-1	7	5
	5-3	6.5	6
	4-4	7	6
	3-6	7	6
	2-7	7	6
	4-3	6	8
	1-3	9	10
Profile #1 Approach Rating	4-C	1.5	2
	2-2	2	2
	2-A	2	3
	5-3	3	2
	4-3	2	5
	7-3	2	6
	2-7	4	3
	4-4	4.5	3
	3-6	5	5
Profile #2 Approach Rating	5-3	4	3
	2-4	5	3
	1-3	6	6

Figure 26a Pilot A - Pilot B Comparison

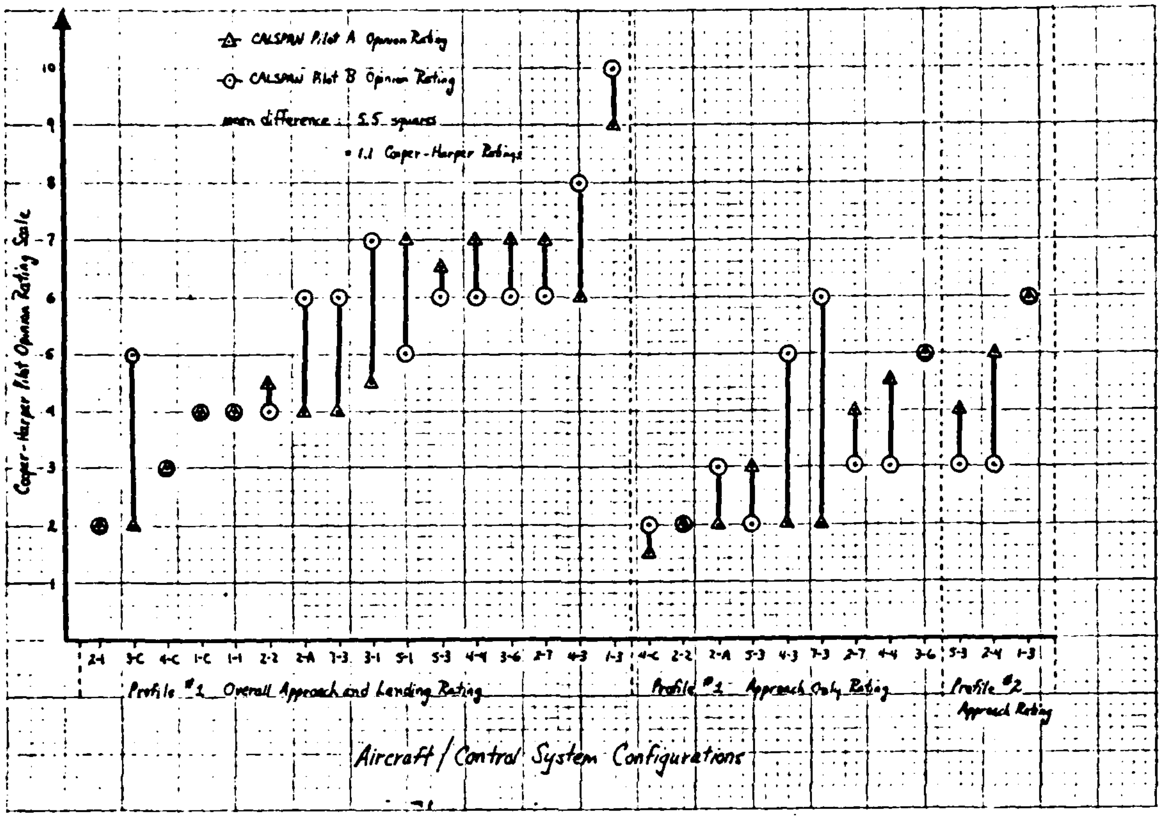


Figure 26b Pilot A - Pilot B Graphical Comparison

evaluations available, however, make such an assumption premature. Also, the effects of arbitrarily selecting the initial conditions is unknown. A second run of OPSACT was made, this time changing the initial conditions to -20 ft in altitude. The results of Run 2 are depicted in Fig. 27 and a comparison of these results to pilot A's ratings is made in Fig. 28. The vertical scales were aligned using the same method used for Run 1. The mean difference in ratings between the model and the pilot increased to 1.61 Cooper-Harper rating increments. Clearly, the use of mean difference in ratings as a means of quantifying the degree of correlation between the

Configuration	Performance Index
2-1	2030
5-3	3483
1-6	2940
2-4	3575
4-4	3479
1-3	2761
3-2	3612
3-3	3430

Initial conditions: -20 ft (altitude)

Figure 27 Optimal Pilot Model Results: Run #2

optimal pilot model and a human pilot's opinion ratings is unreliable unless the effects of the initial conditions are known. Unfortunately, the initial conditions of the CALSPAN evaluations were not known and therefore could not be used in the computer simulations. Since the initial conditions of the optimal pilot model appear to have a significant impact on the rating predictions, it is logical to assume that the model initial conditions would have to be the same as those in the actual flight tests before a comparison between the model results and the pilot ratings could be made. An ... er possibility, though, would be to compare the pilot ratings to ... me result of the optimal pilot model that was independent of the initial conditions.

The state covariance matrix is not dependent on the initial conditions of the model. It is dependent only on the system configuration and its

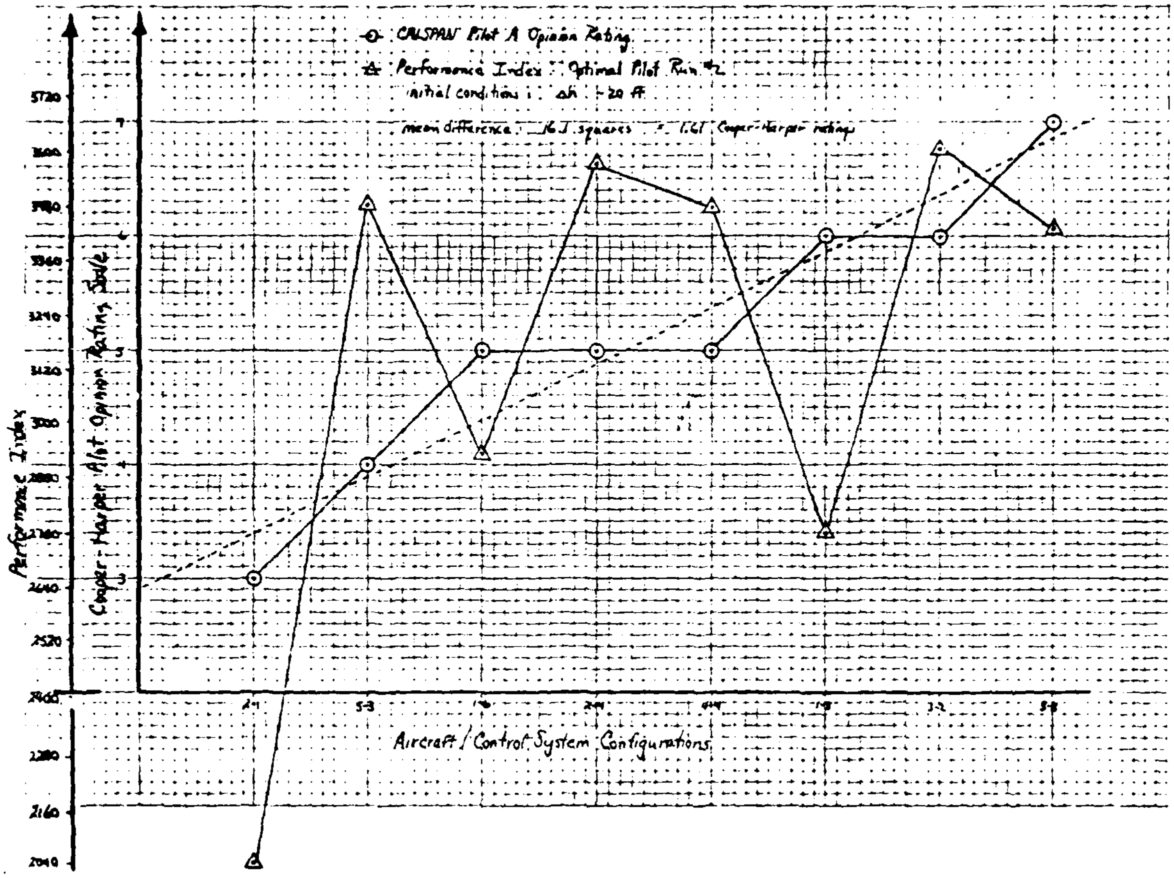


Figure 28 Comparison of Optimal Pilot Model Run #2 Results to CALSPAN Pilot A's Opinion Ratings

response to noise. The diagonal elements of the state covariance matrix (Eq. 27) are the variances of each state. The square root of the variance is the standard deviation, (root mean square of the error about the mean value of the state). Thus, from the state covariance matrix a measure of the perturbations of the states about their nominal values due to system noise can be determined. As applied to this thesis, a system with large values of state variances may respond to system inputs by producing large perturbations in the states. This tendency towards large state perturbations could influence the pilot's opinion of the aircraft's

flying qualities. The RMS error in forward airspeed σ_u, downward airspeed σ_w, pitch rate σ_q, pitch attitude σ_θ, and altitude σ_h were calculated by OPSACT and are listed in Fig. 29. Plots of the errors versus aircraft configuration are depicted in Figs. 30 - 34. Though the errors are different in magnitude, the shapes of the first four plots are quite similar. The plot of the error in altitude is nearly the reverse of the other plots. This may indicate that the system that responds to noise with very small state perturbations will allow the large altitude changes that result from the noise, whereas the system that is highly responsive to the noise will consequentially keep altitude perturbations small.

Figure 35 shows a comparison of the RMS error in pitch response to noise with CALSPAN pilot A's opinion ratings. The verticle scales were aligned in the same manner as with the performance index comparisons. The mean difference in this case was 1 Cooper-Harper rating increment, a difference similar to that of the Pilot A - Pilot B comparison. Also, it can be seen that the RMS error agrees with both performance index comparisons by predicting a much lower (better) rating for Aircraft/Control System Configuration 1-3 than both CALSPAN pilots. While it can be pointed out that the two human pilots disagreed on occasion by a greater amount than the rating difference for Configuration 1-3, and that even with this difference the RMS error predicts pilot A's opinion ratings with an average error of 1 rating increment, the small sample size of this thesis demands a low degree of confidence in this conclusion. Furthermore, a closer look at the comments made by the pilots reveals a lesser amount of correlation between the optimal pilot model predictions and the pilot opinions.

Figure 36 depicts the same results as Fig. 35 but with two additions: CALSPAN pilot B's opinion ratings and a summary of the comments made by

Configuration	σ_u	σ_w	σ_q	σ_θ	σ_h
2-1	11.50	19.44	.1399	.0671	11.29
5-3	11.14	15.34	.0397	.0459	21.45
1-6	11.14	15.81	.0327	.0550	18.55
2-4	11.27	20.20	.1453	.0698	12.59
4-4	11.70	28.58	.2660	.1228	7.60
1-3	11.14	15.81	.0327	.0551	18.58
3-2	12.08	35.06	.3437	.1622	7.54
3-3	12.08	35.09	.3437	.1623	7.56

Figure 29 RMS Error in State Response to System Noise

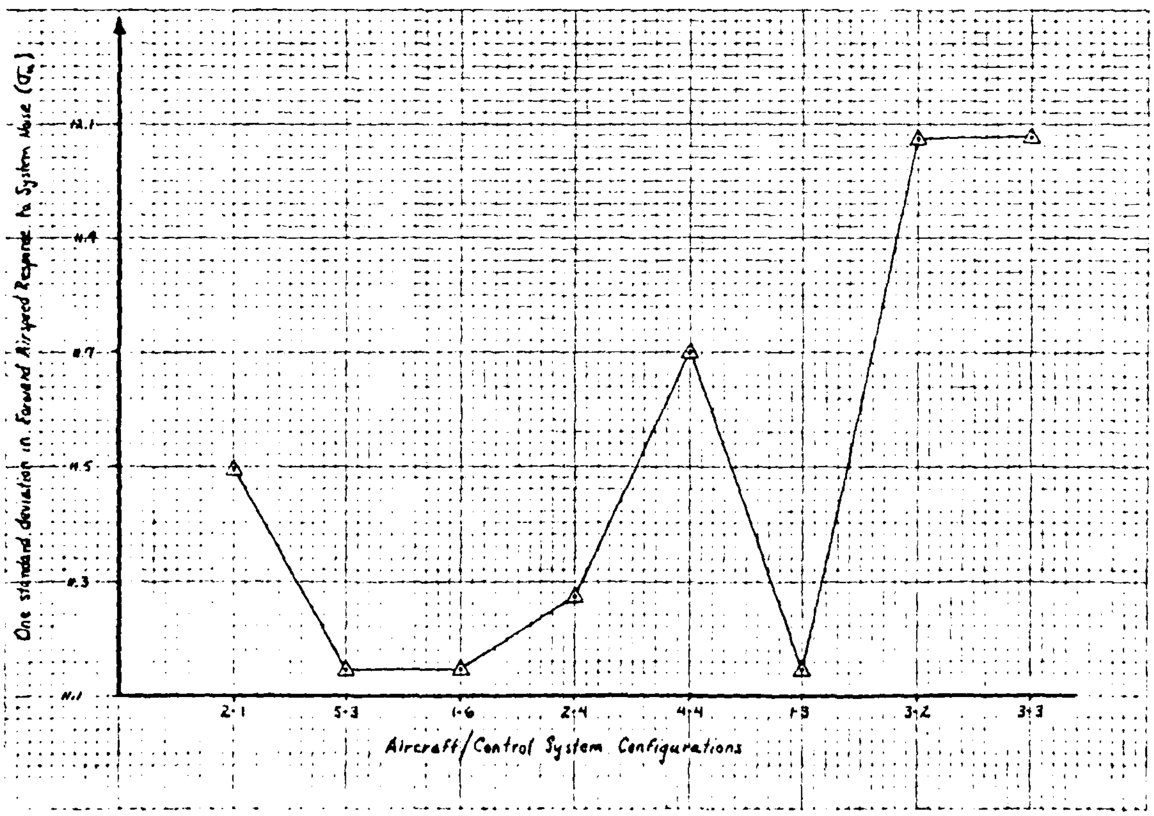


Figure 30 RMS Error in Forward Velocity Response
to System Noise versus Aircraft/Control System Configuration

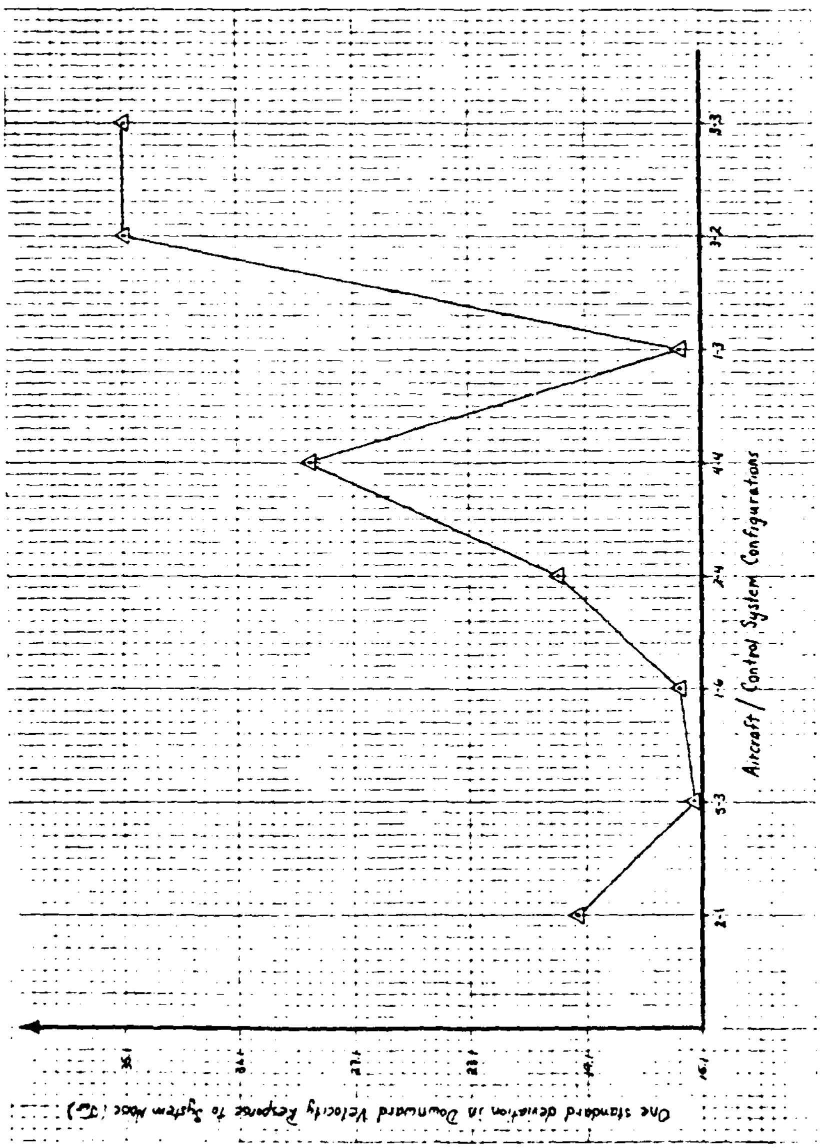


Figure 31 RMS Error in Downward Velocity Response to System Noise versus Aircraft/Control System Configuration

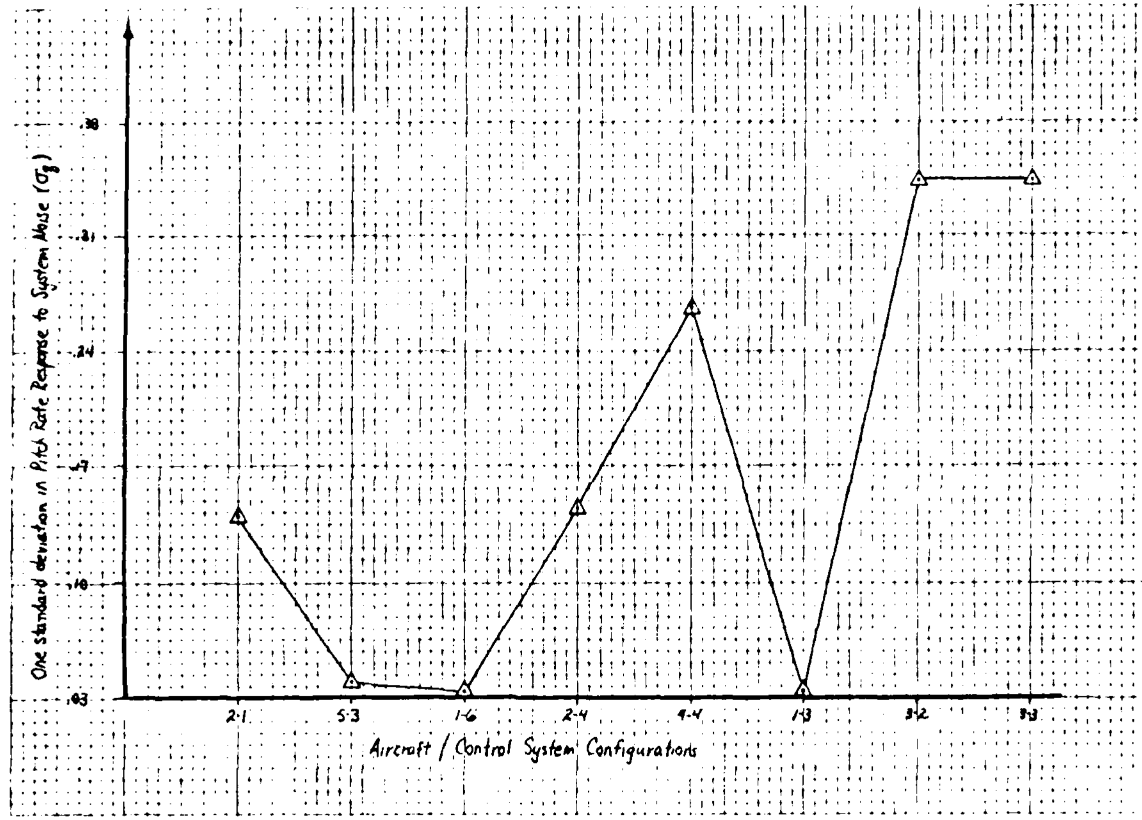


Figure 32 RMS Error in Pitch Rate Response to System Noise versus Aircraft/Control System Configuration

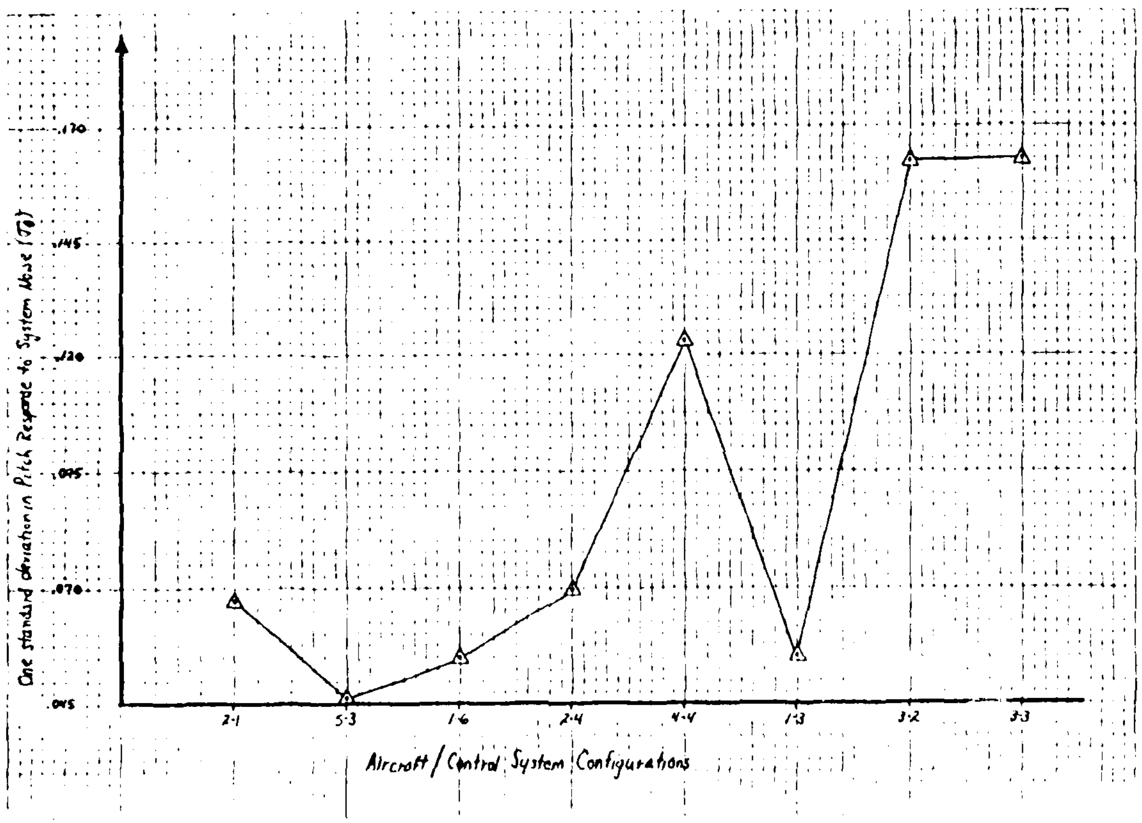


Figure 33 RMS Error in Pitch Attitude Response to System noise versus Aircraft/Control System Configuration

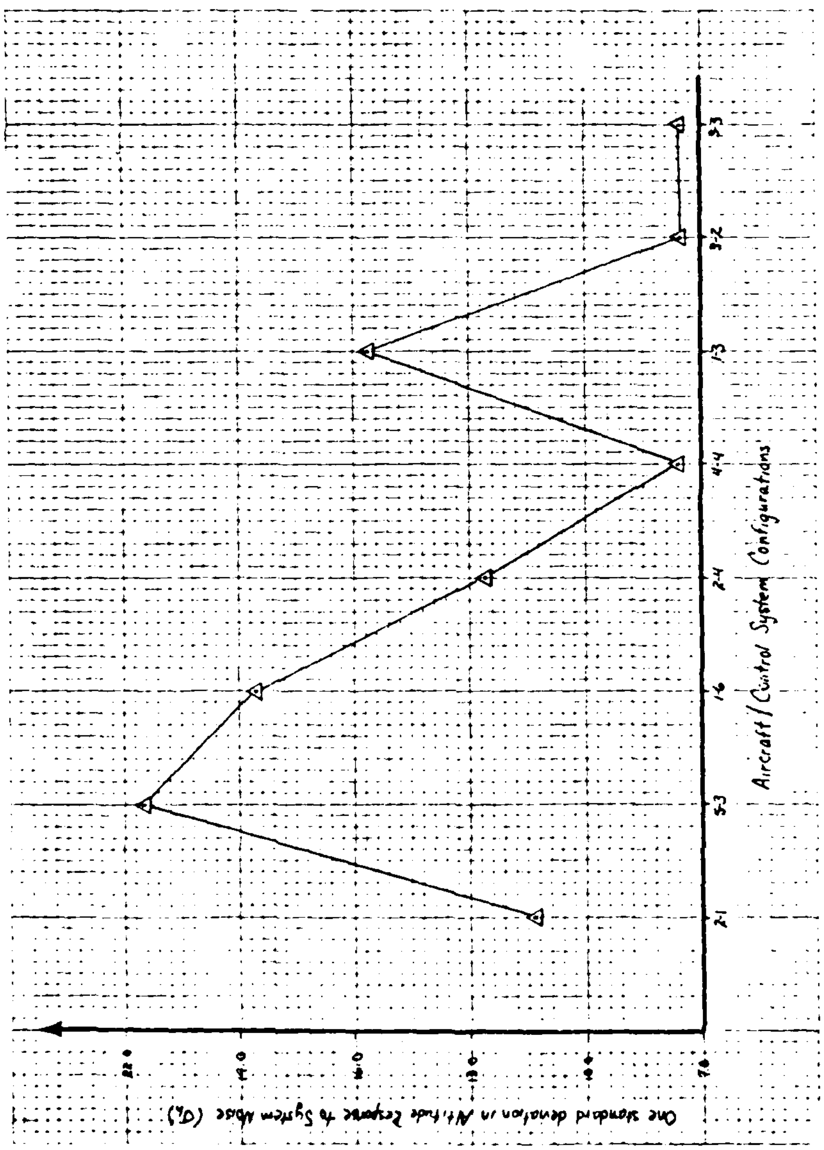


Figure 34 RMS Error In Altitude Response to System Noise versus Aircraft/Control System Configuration

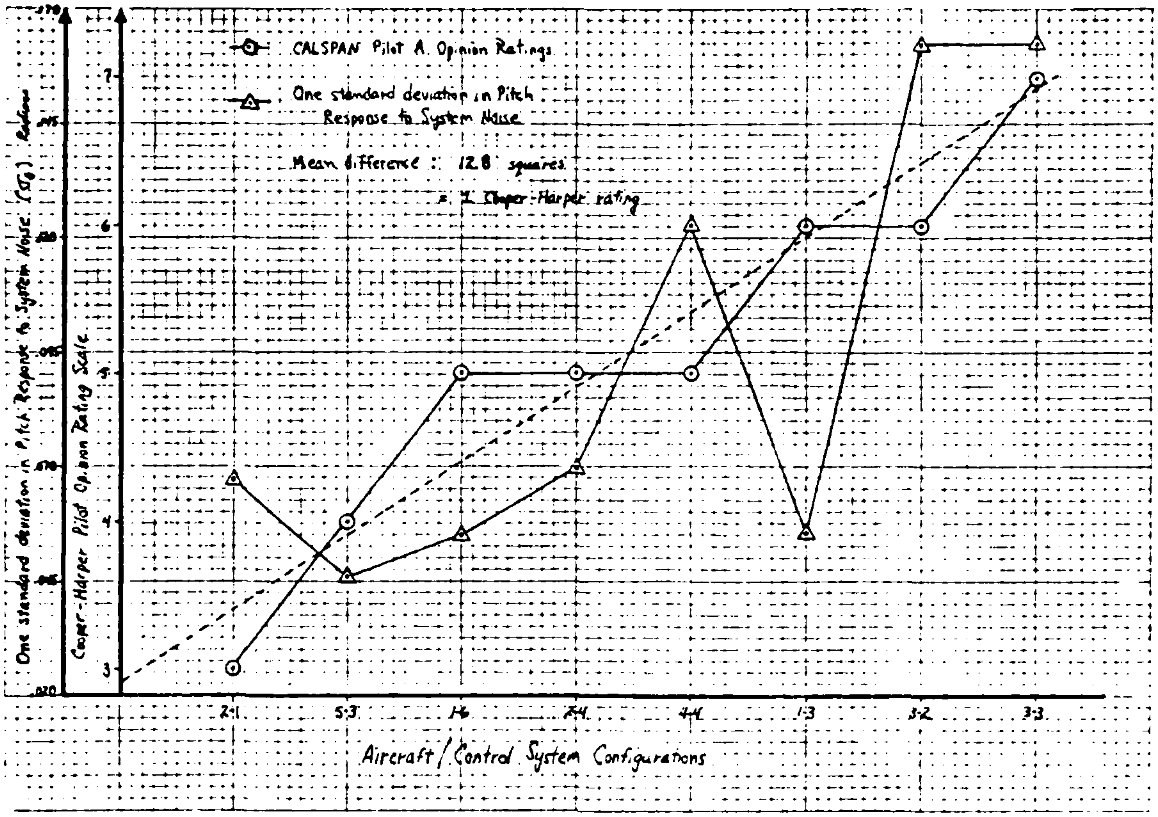


Figure 35 Comparison of RMS Error in Pitch Attitude Response to System Noise with CALSPAN Pilot A's Opinion Ratings

both pilots, (pilot comment sheets in Appendix D). As can be seen from the comments about Configurations 1-3, 3-2, and 3-3, both pilots tend to rate an aircraft with poor pitch response predictability and a tendency towards PIO (Pilot Induced Oscillations) higher than other aircraft (worse rating). The high RMS error for Configurations 3-2 and 3-3 may predict these high ratings, but the low value of RMS error for Configuration 1-3 contradicts this thought. Similarly, pilot A mentioned that there was a tendency to overcontrol Configurations 1-6, 2-4, 4-4, and 1-3. The RMS error for three of these configurations was low and

may have been an indicator of overcontrol problems, but the high RMS error for Configuration 4-4 contradicts this assumption. The problems encountered in trying to find correlation between the optimal pilot model results and the human pilot's opinion ratings (RMS error compared to pilot ratings) are again found in trying to find some connection between the RMS errors and the pilots' comments. The small sample size of this thesis leaves doubt about any conclusion.

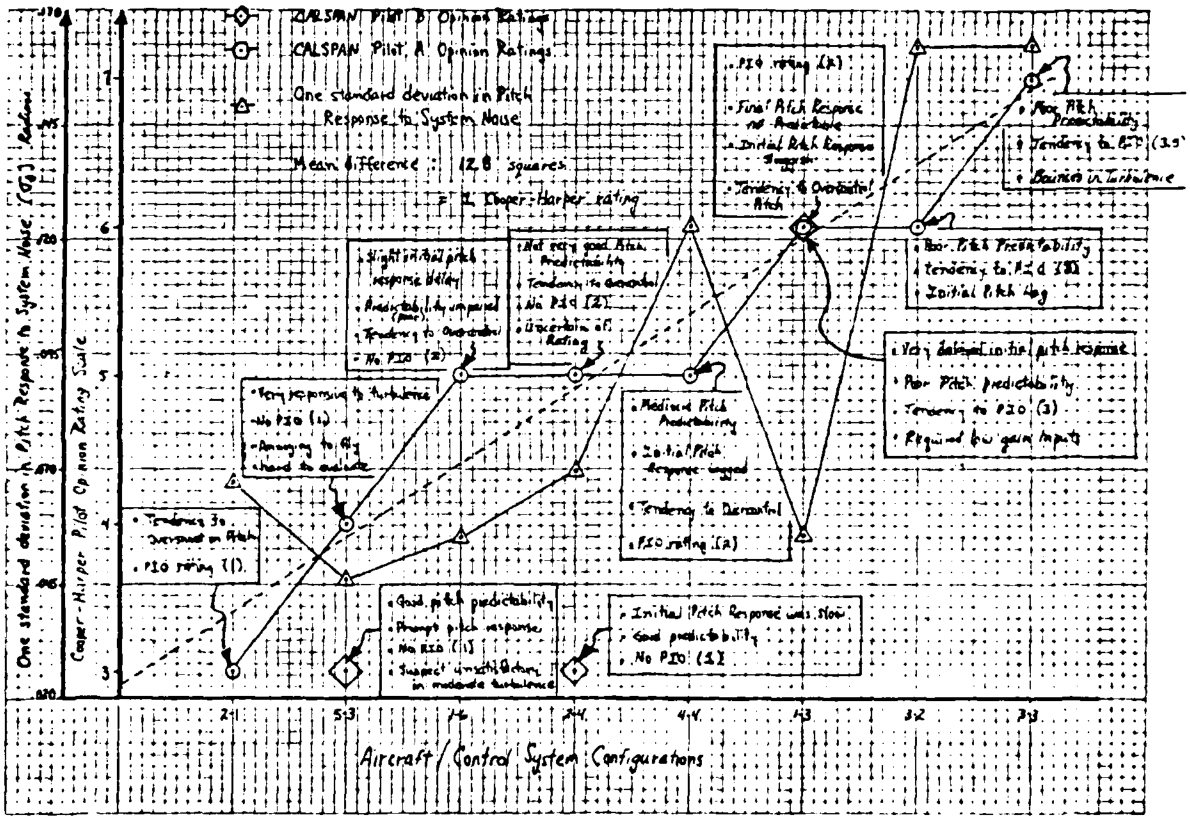


Figure 36 RMS Error - Pilot Rating Comparison with Pilots' Comments

A final attempt to find some method of predicting pilot ratings from the model results was made by plotting the trace of the matrix resulting from the product of the state covariance matrix X and the weighting matrix Q. No definite correlation of any type was observed.

V Conclusions and Recommendations

Due to the small sample size of this thesis, the confidence level for any conclusion made would be extremely low. While some trends, (or perhaps suggestions of trends), can be seen in the results, significant contradictions are also evident. With the scarcity of data available to compare to the optimal pilot model predictions, this thesis has succeeded in primarily pointing to areas where further work work is needed. Still, some things have been learned about possible methods of predicting human pilot opinion ratings based on optimal pilot model results.

Using the performance index as a measure of the aircraft's flying qualities may require that the initial conditions of the model be the same as for the actual flight tests. This doesn't seem reasonable since, first, one would expect that two human pilots would give the same rating to an aircraft even if the initial conditions of the flight tasks were different and, second, if the performance index were an accurate means of predicting flying qualities ratings, it too should give the same rating as a human pilot even if the initial conditions were different. The two CALSPAN pilots did, however, vary significantly in their ratings of several aircraft. The possibility that initial conditions do have a large impact on a pilot's ratings still exists and should be investigated further.

The RMS error in state response to system noise appeared to correlate with CALSPAN pilot A's opinion ratings as well as did pilot B's ratings, though the problem of small sample size generates doubt about the reliability of this correlation. Furthermore, the lack of correlation between the RMS error and the pilots' comments about the aircrafts' flying qualities adds to this doubt. Clearly, more pilot evaluations are needed before anything

definite can be concluded. Unfortunately, calculation of the state covariance matrix (Eq. 27) makes assumptions that limit useable flight evaluations to those using long, constant angle glide slopes without roundout and landing. Evaluations other than those used in this thesis cannot, therefore, be used for comparison with RMS error in state response to system noise. In order to use other CALSPAN evaluations, the optimal pilot model would have to be modified at those points where an infinite glide slope was assumed and provisions would have to be made to account for weighting matrices that change with respect to time and the forcing of the states to their nominal values at the final time. The model would then simulate the entire ILS approach and landing and allow the use of all Profile #1 CALSPAN evaluations for comparison with the model results.

Finally, it should be noted that even if all eighty-three CALSPAN evaluations could be used for comparison to the optimal pilot model results, this data would still be deficient because only two human pilots were used and their debriefings were perhaps a bit too brief. More pilots flying and evaluating the same aircraft/control system configurations are needed in order to determine the mean difference between human pilot ratings. This would indicate how accurate the model predictions actually were. The debriefing of the pilots would have to be much more detailed, including questions on why the pilot gave an aircraft a particular rating. Other questions should be directed towards the aircraft response that results from the RMS error in state response to system noise. Clearly, the use of experimental data not gathered in conjunction with the computer simulations adds many sources of error that can completely distort the results of a study. Still, while this thesis did not show that the optimal pilot model could predict pilot ratings, it did not disprove this theory either.

Bibliography

1. Smith, Rogers E. "Effects of Control Systems Dynamics on Fighter Approach and Landing Longitudinal Flying Qualities, Vol. I", AFFDL-TR 78-122, 1978.

2. Anderson, Ronald O. "A New Approach to the Specification and Evaluation of Flying Qualities", AFFDL-TR-69-120, 1970.

3. Kleinman, David L. and Sheldon Baron. "Analytic Evaluation of Display Requirements for Approach to Landing", NASA CR-1952, 1971.

4. Kleinman, David L. and Sheldon Baron. "A Control Theoretic Model for Pilotted Approach to Landing", Automatica, Vol. 9, 1973, pp. 339-347.

5. Kleinman, David L. and Sheldon Baron. "Application of Optimal Control Theory to the Prediction of Human Performance in a Complex Task", AFFDL-TR-69-81, 1970.

6. Kleinman, David L., Sheldon Baron, and William H. Levison. "An Optimal Control Model for Human Response - Part I: Theory and Validation", Automatica, Vol. 6, 1970, pp. 357-369.

7. Hess, Ronald A. "Prediction of Pilot Opinion Ratings Using an Optimal Pilot Model", Human Factors, Vol 19, No. 5, Oct 1977, pp 459-476.

8. Rynaski, Edmund G. and Richard F. Whitbeck. "The Theory and Application of Linear Optimal Control", AFFDL-TR-65-28, 1966.

9. Silverthorn, James . Class notes, MC-729, Air Force Institute of Technology, 1980.

10. Levison, William H., Sheldon Baron, and David L. Kleinman. "A Model for Human Controller Remnant", IEEE Transactions on Man-Machine Systems, MMS-10, No. 4, 1969.

11. Kleinman, David L. "Optimal Control of Linear Systems with Time-Delay and Observation Noise", IEEE Transactions on Automatic Control, AC-14, 1969.

12. Kuo, Benjamin C. Digital Control Systems. SRL Publishing Co., 1977.

13. Fortman, Thomas E. and Konrad L. Hitz. An Introduction to Linear Control Systems, Marcel Dekker, Inc. 1977.

14. Kwakernaak, Huibert and Raphael Sivan. Linear Optimal Control Systems, John Wiley and Sons, Inc., 1972.

15. Kleinman, David L. "Computer Programs Useful in Linear Systems Studies", SCI Technical Memorandum, Systems Control, Inc., Palo Alto, Calif., 1971.

Appendix A

First Necessary Conditions for Optimality

The purpose of an optimal control system is to find the input $\underline{u}^*(t)$ that minimizes the performance index

$$J(u) = f_1[\underline{x}(t_f)] + \int_{t_o}^{t_f} f_2[\underline{x}(t),\underline{u}(t),t]dt \qquad \text{(A-1)}$$

subject to the constraining equation

$$\dot{\underline{x}}(t) = f_3[\underline{x}(t),\underline{u}(t),t] \ ; \quad \underline{x}(t_o) = \underline{x}_o \qquad \text{(A-2)}$$

One method for developing the First Necessary Conditions for Optimality is described by Silverthorn (Ref. 9). The state variables $\underline{x}(t)$ and the control inputs $\underline{u}(t)$ are treated as being independent by introducing a vector of Lagrange multipliers $\underline{\boldsymbol{\lambda}}(t)$ to augment the performance index with the constraining equation

$$J(u) = f_1[\underline{x}(t_f)] + \int_{t_o}^{t_f} f_2[\underline{x}(t),\underline{u}(t),t]$$

$$+ \underline{\boldsymbol{\lambda}}^T(t)[f_3[\underline{x}(t),\underline{u}(t),t] - \dot{\underline{x}}(t)] \qquad \text{(A-3)}$$

The Hamiltonian is defined by

$$H[\underline{x}(t),\underline{u}(t),t] = f_2[\underline{x}(t),\underline{u}(t),t]$$

$$+ \underline{\lambda}^T(t)[f_3[\underline{x}(t),\underline{u}(t),t]] \tag{A-4}$$

Equation A-3 becomes

$$J(u) = f_1[\underline{x}(t_f)]$$

$$+ \int_{t_o}^{t_f} \left\{ H[\underline{x}(t),\underline{u}(t),\underline{\lambda}(t),t] - \underline{\lambda}^T(t)\dot{\underline{x}}(t) \right\} dt \tag{A-5}$$

Recalling that $\underline{u}^*(t)$ is defined as the control input that minimizes $J(u)$, then $\underline{x}^*(t)$ and $\underline{\lambda}^*(t)$ represent the corresponding optimal trajectories. Non-optimum values of these variables separated from the optimum by a small variation are represented by

$$\underline{u}(t) = \underline{u}^*(t) + \epsilon\,\underline{\eta}(t) \tag{A-6}$$

$$\underline{x}(t) = \underline{x}^*(t) + \epsilon\,\underline{\xi}(t) \tag{A-7}$$

$$\underline{\lambda}(t) = \underline{\lambda}^*(t) + \epsilon\,\underline{\gamma}(t) \tag{A-8}$$

Substituting the relations in Eqs. A-6, A-7, and A-8 into Eq. A-5 and expanding in a Taylor series yields Eq. A-9 on the following page. The First Necessary Conditions for Optimality are based on the following argument. By choosing ϵ arbitrarily small, the ϵ term in Eq. A-9, unless it is zero, will overwhelm the higher order terms. Since ϵ can be chosen either positive or negative, $J(u)$ evaluated at some non-optimum $\underline{u}(t)$ will

$$
\begin{aligned}
J(u) &= f_1[\underline{x}^*(t_f) + \epsilon\underline{\xi}(t_f)] + \int_{t_o}^{t_f} \Big\{ H[\underline{x}^*(t) + \epsilon\underline{\xi}(t), \underline{u}^*(t) + \epsilon\underline{\eta}(t), \underline{\lambda}^*(t) + \epsilon\underline{\gamma}(t)] \\
&\qquad - [\underline{\lambda}^{*T}(t) + \epsilon\underline{\gamma}^T(t)][\dot{\underline{x}}^*(t) + \epsilon\dot{\underline{\xi}}(t)] \Big\} dt \\
&= f_1[\underline{x}^*(t_f)] + \left(\frac{\partial f_1}{\partial \underline{x}}\right)^T \Big|_{*,t_f} \epsilon\underline{\xi}(t_f) + \int_{t_o}^{t_f} \Big\{ H[\underline{x}^*(t), \underline{u}^*(t), \underline{\lambda}^*(t), t] + \left(\frac{\partial H}{\partial \underline{x}}\right)^T \Big|_* \epsilon\underline{\xi}(t) \\
&\qquad + \left(\frac{\partial H}{\partial \underline{u}}\right)^T \Big|_* \epsilon\underline{\eta}(t) + \left(\frac{\partial H}{\partial \underline{\lambda}}\right)^T \Big|_* \epsilon\underline{\gamma}(t) - [\underline{\lambda}^{*T}(t)\dot{\underline{x}}^*(t) + \underline{\lambda}^{*T}(t)\epsilon\dot{\underline{\xi}}(t) \\
&\qquad + \epsilon\underline{\gamma}^T(t)\dot{\underline{x}}^*(t)] \Big\} dt + O(\epsilon^2) \\
&= J(u^*) + \epsilon \Big\{ \left(\frac{\partial f_1}{\partial \underline{x}}\right)^T \Big|_{*,t_f} \underline{\xi}(t_f) + \int_{t_o}^{t_f} [\left(\frac{\partial H}{\partial \underline{x}}\right)^T \Big|_* \underline{\xi}(t) + \left(\frac{\partial H}{\partial \underline{u}}\right)^T \Big|_* \underline{\eta}(t) + \left(\frac{\partial H}{\partial \underline{\lambda}}\right)^T \Big|_* \underline{\gamma}(t) \\
&\qquad - \underline{\lambda}^{*T}(t)\dot{\underline{\xi}}(t) - \underline{\gamma}^T(t)\dot{\underline{x}}^*(t)] dt \Big\} + O(\epsilon^2)
\end{aligned}
\tag{A-9}
$$

be greater than or equal to $J(u^*)$ only if the coefficient of the term is zero. Thus, for $\underline{u}^*(t)$ to be an optimal control, the following must be true:

$$\left\{ \left(\frac{\partial f_1}{\partial \underline{x}}\right)^T\Bigg|_{*,t_f} \underline{\xi}(t) + \int_{t_o}^{t_f} \left(\frac{\partial H}{\partial \underline{x}}\right)^T\Bigg|_{*} \underline{\xi}(t) + \left(\frac{\partial H}{\partial \underline{u}}\right)^T\Bigg|_{*} \underline{\eta}(t) + \left(\frac{\partial H}{\partial \underline{\lambda}}\right)^T\Bigg|_{*} \underline{\gamma}(t) - \underline{\lambda}^{*T}(t)\dot{\underline{\xi}}(t) - \underline{\gamma}^T(t)\dot{\underline{x}}^*(t) \right\} = 0 \qquad \text{(A-10)}$$

The values $\underline{\xi}(t)$ and $\dot{\underline{\xi}}(t)$ cannot be chosen independently, so by integrating the $\dot{\underline{\xi}}(t)$ term by parts and rearranging, Eq. A-10 becomes

$$\left\{ \left(\frac{\partial f_1}{\partial \underline{x}}\right)^T\Bigg|_{*,t_f} - \underline{\lambda}^{*T}(t_f) \right\} \underline{\xi}(t_f) + \int_{t_o}^{t_f} \left\{ \left(\frac{\partial H}{\partial \underline{x}}\right)^T\Bigg|_{*} + \dot{\underline{\lambda}}^{*T}(t) \right\} \underline{\xi}(t)dt + \int_{t_o}^{t_f} \left(\frac{\partial H}{\partial \underline{u}}\right)^T\Bigg|_{*} \underline{\eta}(t)dt + \int_{t_o}^{t_f} \left\{ \left(\frac{\partial H}{\partial \underline{\lambda}}\right)^T\Bigg|_{*} - \dot{\underline{x}}^{*T}(t) \right\} \underline{\gamma}(t)dt = 0 \qquad \text{(A-11)}$$

Since the selection of $\underline{\xi}(t)$ is arbitrary, one choice is to select $\underline{\xi}(t_f) = 0$. Each of the remaining integrals in Eq. A-11 must be equal to zero. Applying the Fundamental Lemma of the Calculus of Variations to the integrals yields

$$\left(\frac{\partial H}{\partial \underline{x}}\right)^T\Big|_* + \dot{\underline{\lambda}}^{*T}(t) = 0 \qquad \text{(A-12)}$$

$$\left(\frac{\partial H}{\partial \underline{u}}\right)^T\Big|_* = 0 \qquad \text{(A-13)}$$

$$\left(\frac{\partial H}{\partial \underline{\lambda}}\right)^T\Big|_* - \dot{\underline{x}}^{*T}(t) = 0 \qquad \text{(A-14)}$$

Another choice for $\underline{\xi}(t)$ is $\underline{\xi}(t_f) \neq 0$. Thus, the first term in Eq. A-11 becomes

$$\left(\frac{\partial f_1}{\partial \underline{x}}\right)^T\Big|_{*,t_f} - \underline{\lambda}^{*T}(t_f) = 0 \qquad \text{(A-15)}$$

Also, recall that

$$\underline{x}(t_o) = \underline{x}_o \qquad \text{(A-16)}$$

Equations A-12, A-13, A-14, A-15, and A-16 are the First Necessary Conditions for Optimality. The general optimal control problem could now be solved, except that the boundary conditions represented by Eqs. A-15 and A-16 are "split", (one boundary condition is given at the initial time and the other at the final time). The solution must be guessed at, compared to the First Necessary Conditions for Optimality, and then iteratively reguessed.

The exception to the iterative solution process occurs when the

system is linear and the performance index is a quadratic form. The general form of the linear system, quadratic performance index is

$$J(u) = \tfrac{1}{2}\, \underline{x}^T(t_f) H \underline{x}(t_f) + \tfrac{1}{2} \int_{t_o}^{t_f} [\underline{x}^T(t) Q \underline{x}(t) + \underline{u}^T(t) R \underline{u}(t)]dt \tag{A-17}$$

The Hamiltonian is

$$H[\underline{x}(t), \underline{u}(t), \underline{\lambda}(t), t] = \tfrac{1}{2}\, \underline{x}^T(t) Q \underline{x}(t) + \tfrac{1}{2}\, \underline{u}^T(t) R \underline{u}(t) + \underline{\lambda}^T(t)[A\underline{x}(t) + B\underline{u}(t)] \tag{A-18}$$

where the constraining equation is

$$\dot{\underline{x}}(t) = A\underline{x}(t) + B\underline{u}(t) \;;\quad \underline{x}(t_o) = \underline{x}_o \tag{A-19}$$

Applying the First Necessary Conditions for Optimality yields

$$\dot{\underline{\lambda}}^*(t) = - \left.\frac{\partial H}{\partial \underline{x}}\right|_* = - Q\underline{x}^*(t) - A^T \underline{\lambda}^*(t) \tag{A-20}$$

$$0 = R\underline{u}^*(t) + B^T \underline{\lambda}^*(t) \tag{A-21}$$

$$\dot{\underline{x}}(t) = A\underline{x}(t) + B\underline{u}(t) \tag{A-22}$$

$$\underline{\lambda}^*(t_f) = H\underline{x}^*(t_f)$$

AD-A111 136 AIR FORCE INST OF TECH WRIGHT-PATTERSON AFB OH SCHOO--ETC F/G 5/8
PREDICTING PILOT OPINION RATINGS OF FLYING QUALITIES OF HIGHLY --ETC(U)
DEC 81 R M ENRIGHT
UNCLASSIFIED AFIT/GAE/AA/80D-3 NL

2 OF 2
AD A
111136

END
DATE
FILMED
03-82
DTIC

$$\underline{x}(t_o) = \underline{x}_o \qquad \text{(A-24)}$$

Eq. A-21 becomes

$$\underline{u}^*(t) = -R^{-1}B^T\underline{\lambda}^*(t) \qquad \text{(A-25)}$$

Combining Eqs. A-20, A-22, and A-25 into matrix form yields

$$\begin{bmatrix} \dot{\underline{x}}(t) \\ \hline \dot{\underline{\lambda}}(t) \end{bmatrix}_* = \left[\begin{array}{c|c} A & -BR^{-1}B^T \\ \hline -Q & -A^T \end{array}\right] \begin{bmatrix} \underline{x}(t) \\ \hline \underline{\lambda}(t) \end{bmatrix}_* \qquad \text{(A-26)}$$

The expression for $\underline{\lambda}(t)$ is assumed to be a linear combination of $\underline{x}(t)$; thus,

$$\underline{\lambda}(t) = K(t)\underline{x}(t) \qquad \text{(A-27)}$$

Eq. A-26 becomes

$$\begin{bmatrix} \underline{x}(t) \\ \hline K(t)\dot{\underline{x}}(t) + \dot{K}(t)\underline{x}(t) \end{bmatrix} = \left[\begin{array}{c|c} A & -BR^{-1}B^T \\ \hline -Q & -A^T \end{array}\right] \begin{bmatrix} \underline{x}(t) \\ \hline K(t)\underline{x}(t) \end{bmatrix} \qquad \text{(A-28)}$$

Reducing Eq. A-28 to a single equation yields

$$-\dot{K}(t) = K(t)A + A^TK(t) + Q - K(t)BR^{-1}B^TK(t) \qquad \text{(A-29)}$$

Eqs. A-27 and A-23 imply that

$$K(t_f) = H \tag{A-30}$$

and combining Eqs. A-25 and A-27 yield

$$\underline{u}^*(t) = -R^{-1}B^TK(t)\underline{x}(t) \tag{A-31}$$

If the final time is far off into the future relative to the time constants of the system, K(t) can be considered a constant. Thus, the weighting on the final states is irrelevant and Eq. A-29 becomes

$$0 = KA + A^TK + Q - KBR^{-1}B^TK \tag{A-32}$$

and Eq. A-31 becomes

$$\underline{u}^*(t) = -R^{-1}B^TK\underline{x}(t) \tag{A-33}$$

Appendix B

Mathematical Development of the Estimator and the Predictor

As mentioned in Chapter II, the optimal pilot model evolves from the optimal control system through the addition of an estimator and a predictor. The mathematical development of these two components is now discussed.

Estimator

Since the system states cannot be directly observed by the pilot, they must be reconstructed in his mind based on some known information. His estimation of what the states are is modeled using a full-state observer. The observer reconstructs the system states $\underline{x}(t)$, which are defined by

$$\dot{\underline{x}}(t) = A_1\underline{x}(t) + B_1\underline{u}(t) \; ; \; \underline{x}(t_o) = \underline{x}_o \qquad \text{(B-1)}$$

and produces an estimate of the states $\underline{x}(t)$ according to

$$\dot{\hat{\underline{x}}}(t) = A_1\hat{\underline{x}}(t) + B_1\underline{u}(t) \; ; \; \hat{\underline{x}}(t_o) = \hat{\underline{x}}_o \qquad \text{(B-2)}$$

Using the same input $\underline{u}(t)$ as the system and setting $\hat{\underline{x}}_o = \underline{x}_o$, the observer should provide an exact estimate $\hat{\underline{x}}(t) = \underline{x}(t)$. However, due to noise, modeling inaccuracies, and initial conditions an exact estimate is not possible (Ref. 13). To improve the estimate, more information about the system is available to the pilot; namely, the displayed variables. Recall that

$$\underline{y}(t) = C\underline{x}(t) \tag{B-3}$$

The estimate of $\underline{y}(t)$ is

$$\hat{\underline{y}}(t) = C\hat{\underline{x}}(t) \tag{B-4}$$

The error in estimating the displayed variables is

$$\underline{y}_e(t) = \underline{y}(t) - \hat{\underline{y}}(t) \tag{B-5}$$

To account for modeling errors and noise, a correction term is added to Eq. B-2

$$\dot{\hat{\underline{x}}}(t) = A_1\hat{\underline{x}}(t) + B_1\underline{u}(t) + S[\underline{y}_e(t)] \tag{B-6}$$

where S is the estimator gain matrix and will be discussed in more detail later. By defining the error in estimating the states as

$$\underline{x}_e(t) = \underline{x}(t) - \hat{\underline{x}}(t) \tag{B-7}$$

it can be seen that

$$\dot{\underline{x}}_e(t) = \dot{\underline{x}}(t) - \dot{\hat{\underline{x}}}(t) = A_1\underline{x}_e(t) - S[\underline{y}_e(t)]$$

$$= (A_1 - SC)\underline{x}_e(t) \tag{B-8}$$

or equivalently

$$\underline{x}_e(t) = e^{(A_1-SC)(t-t_o)}\underline{x}_e(t_o) \qquad \text{(B-9)}$$

The estimate error will decay to zero only if S is chosen such that the system described by Eq. B-8 is stable (Refs. 13,14). Assuming complete observ-bility of the system, the S matrix can be chosen such that the observer poles defined by the eigenvalues of the matrix $[A_1-SC]$ are far enough into the left-hand complex plane to insure that the decay of the error is fast enough, yet not so far to the left as to make the S matrix large and the observer very sensitive to observation noise (Ref. 14).

Before determining the gain matrix S, it should be recalled from Chapter II that the system is actually a stochastic process. Random variables are introduced into the system; therefore, Eqs. B-1 and B-3 are rewritten as

$$\dot{\underline{\tilde{x}}}(t) = A_1\underline{\tilde{x}}(t) + B_1\underline{u}(t) + D_2w_{g_1}(t) \qquad \text{(B-10)}$$

$$\underline{y}(t) = C\underline{\tilde{x}}(t) + \underline{v}_y(t) \qquad \text{(B-11)}$$

where $w_{g_1}(t)$ and D_2 are as described by Eq. 24, and $\underline{v}_y(t)$ represents observation noise. Both processes are assumed to be zero-mean, white Gaussian noise of intensities W_1 and V_y, respectively. The optimal observer, as defined by Eq. B-6, attempts to minimize the mean square error

$$E\left\{\underline{x}_e^T(t)G\underline{x}_e(t)\right\} = \bar{\underline{x}}_e^T(t)\,G\bar{\underline{x}}_e(t) + tr[PG] \qquad \text{(B-12)}$$

where G is a symmetric, positive definite weighting matrix, $\bar{\underline{x}}_e(t)$ is the mean of $\underline{x}_e(t)$, and P is the estimator error covariance matrix (Ref. 14).

$$P(t) = E\left\{[\underline{x}_e(t) - \bar{\underline{x}}_e(t)][\underline{x}_e(t)-\bar{\underline{x}}_e(t)]^T\right\} \tag{B-13}$$

As mentioned before, apart from the effects of noise, model inaccuracies, etc., the estimator error can be minimized by the proper choice of the initial conditions. The first term of Eq. B-12 is minimized when $\hat{\underline{x}}_o = \underline{x}_o$. The stochastic inputs in Eqs. B-10 and B-11 change the estimator error described by Eq. B-8 to

$$\begin{aligned}\dot{\tilde{\underline{x}}}_e(t) &= (A_1-SC)\tilde{\underline{x}}_e(t) + D_2 w_{g_1}(t) - S\underline{v}_y(t) \\ &= (A_1-SC)\tilde{\underline{x}}_e(t) + F\begin{bmatrix} w_{g_1}(t) \\ \hline v_y(t)\end{bmatrix}\end{aligned} \tag{B-14}$$

where $F = \left[D_2 \;\vdots\; -S\right]$.

In this study, the estimator is assumed to have been operating for some time beyond the start of the flight. Therefore, it can be assumed that the estimator error covariance matrix P has reached a steady state value, (much the same way the solution to the optimal control Riccati equation K can be considered to be constant if the final time is far into the future). Thus, it can be shown (Ref. 14) that the steady state estimator error covariance matrix is

$$P = \int_0^{\infty} e^{(A_1 - SC)t} FW_2F^T e^{(A_1-SC)^T t} dt, \quad W_2 = \left[\begin{array}{c|c} W_1 & 0 \\ \hline 0 & V_y \end{array}\right] \tag{B-15}$$

The covariance matrix P satisfies the equation

$$0 = (A_1 - SC)P + P(A_1 - SC)^T + FW_2F^T$$

$$= (A_1 - SC)P + P(A_1 - SC)^T + W_1 + SV_yS^T \tag{B-16}$$

Furthermore, P is minimized if (Ref. 14)

$$S = PC^TV_y^{-1} \tag{B-17}$$

Substituting Eq. B-17 into Eq. B-16 yields

$$0 = A_1P + PA_1^T + W_1 - PC^TV_y^{-1}CP \tag{B-18}$$

The value of P, when applied to Eq. B-17, yields an estimator gain matrix S that minimizes the second term in Eq. B-12. Figure 37 shows how the estimator fits into the optimal pilot model.

Predictor

When a delay τ exists in the system, the estimator produces the delayed estimate of the states $\hat{\underline{x}}(t-\tau)$, instead of $\hat{\underline{x}}(t)$. Eq. B-6 becomes

$$\dot{\hat{\underline{x}}}(t-\tau) = A_1\hat{\underline{x}}(t-\tau) + B_1\underline{u}(t-\tau) + S[\underline{y}(t-\tau) - C\hat{\underline{x}}(t-\tau)] \tag{B-19}$$

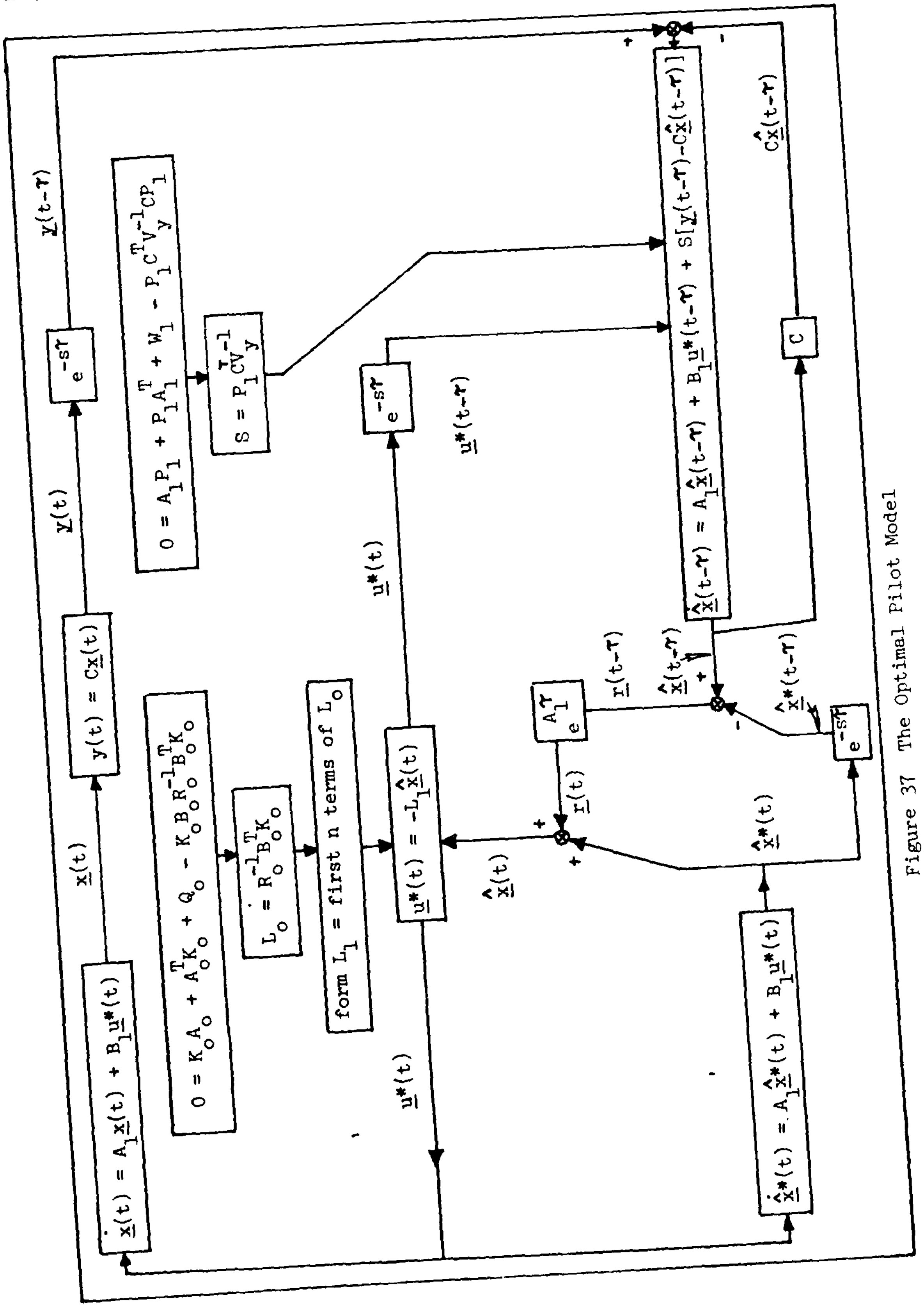


Figure 37 The Optimal Pilot Model

The predictor models the pilot's reconstruction of the system states $\hat{\underline{x}}(t)$ based on the estimate of the delayed states $\hat{\underline{x}}(t-\tau)$. The predictor is developed by Kleinman (Ref. 11) by first defining $\hat{\underline{x}}^*(t)$ as the part of $\hat{\underline{x}}(t)$ due to the deterministic input $\underline{u}^*(t)$, where $\hat{\underline{x}}^*(t)$ satisfies the relation

$$\dot{\hat{\underline{x}}}^*(t) = A_1\hat{\underline{x}}^*(t) + B_1\underline{u}^*(t) \tag{B-20}$$

The predictor error is then defined as

$$\underline{r}(t) = \hat{\underline{x}}(t) - \hat{\underline{x}}^*(t) \tag{B-21}$$

which is a random process and evolves according to

$$\dot{\underline{r}}(t) = A_1\underline{r}(t) + S[\underline{y}(t)-C\hat{\underline{x}}(t)] \tag{B-22}$$

Eq. 21 can be written as

$$\begin{aligned}\hat{\underline{x}}(t) &= \hat{\underline{x}}^*(t) + \underline{r}(t)\\ &= \hat{\underline{x}}^*(t) + E\left\{\underline{r}(t)\middle|\hat{\underline{x}}(\sigma),\ \sigma\leq t\right\}\\ &= \hat{\underline{x}}^*(t) + E\left\{\underline{r}(t)\middle|\hat{\underline{x}}(\sigma),\ \sigma\leq t-\tau\right\}\end{aligned} \tag{B-23}$$

Since $\underline{r}(t)$ is independent of $\underline{u}^*(t)$, Eq. B-23 becomes

$$\begin{aligned}\hat{\underline{x}}(t) &= \hat{\underline{x}}^*(t) + E\left\{\underline{r}(t)\middle|\underline{r}(\sigma),\sigma\leq t-\tau\right\}\\ &= \hat{\underline{x}}^*(t) + e^{A_1\tau}\,\underline{r}(t-\tau)\end{aligned} \tag{B-24}$$

The reconstructed state estimate is determined by Eq. B-24. Figure 37 shows how the predictor is added to the optimal pilot model.

Appendix C

Optimal Pilot - Single Axis Control Task (OPSACT) Program Listing

```
      PROGRAM OPSACT (INPUT=/80,OUTPUT,PLOT,TAPE5=INPUT,TAPE6=OUTPUT)
C
C         PROGRAM OPSACT (OPTIMAL PILOT - SINGLE AXIS CONTROL TASK),
C     AT THIS STAGE OF ITS DEVELOPMENT, IS DESIGNED TO MODEL THE
C     OPTIMAL PILOT PERFORMING AN INFINITE TIME LINEAR REGULATOR TASK.
C     THE INPUT VALUES ARE:
C         (1) SYSTEM COEFFICIENT MATRIX (AO)
C         (2) CONTROL INPUT MATRIX (BO)
C         (3) OUTPUT MEASUREMENT MATRIX (C)
C         (4) STATE WEIGHTING MATRIX (QO)
C         (5) CONTROL WEIGHTING MATRIX (RO)
C         (6) OBSERVATION NOISE COVARIANCE MATRIX (V)
C         (7) EXTERNAL DISTURBANCE COVARIANCE MATRIX (WO)
C         (8) INITIAL TIME (TI)
C         (9) FINAL TIME (TF)
C        (10) INTEGRATION TIME INCREMENT (TINCR)
C        (11) INITIAL STATE VALUES (XI)
C        (12) INITIAL CONTROL INPUT VALUES (UI)
C        (13) INSTRUMENT THRESHOLDS
C
C         THE SYSTEM IS PRESENTLY LIMITED TO A TWELFTH ORDER PROBLEM.
C
      REAL A(12,12),B(12,12),C(12,12),Q(12,12),V(12,12),R(12,12),
     :     W(12,12),P(12,12),K(12,12),S(12,12),I(12,12),PHI(12,12),
     :     QREV(12,12),GAMMA(12,12),PSI(12,12),RESULT(12,12),
     :     RINV(12,12),BTRANS(12,12),RESULT2(12,12),COUT(12,12),
     :     CTRANS(12,12),VINV(12,12),ATRANS(12,12),TAUN(12,12),
     :     U2(12,12),X2(12,12),Z(12,12),Z1(12,12),RESULT3(12,12),
     :     XDELAY(12,12),UDELAY(12,12),YDELAY(12,12),
     :     XNEW(12),UOLD(12),DXHATN(12),DXHATO(12),Y(12),
     :     YEST(12),DEE(12),EE(12),XHAT(12),OXHATN(12),UOPT(12),
     :     OUTPUT(12),Z2(12),Z3(12),Z4(12),THRSHD(12),
     :     XOLD(12),OXHATO(12),XI(12),UI(12),TP(300),YA(1200),
     :     DDXHAT(12),DUOPT(12),YNEW(12),
     :     TOLD,TINCR,TNEW,TF,TOL,TAU,TI,DT,U3,X3,PFM,PFMT,SUM1,SUM2
C
      INTEGER MODE,N,M,YN,IER,J,MR,IRVING,CHANGE,I,II,INDEX,NLEN,
     :        NOUT,COUNT,MORE,JJ,LA,INC,ID,IE,IG,NPOINTS,LIMIT,RUN,
     :        EIG,GUESS,MM,III
C
      CHARACTER HANDLE*7
C
      COMMON/MAIN1/NDIM,NDIM1,DUM(144)
      COMMON/MAIN2/STORE(144)
      COMMON/MAIN3/EXTRA(144)
      COMMON/INOU/KIN,KOUT,KPU
      COMMON/GRAFS/TITLE(3),Y1TITLE(2),Y2TITLE(2),Y3TITLE(2),Y4TITLE(2)
      COMMON/BOMB/EIG
C
C
      CHANGE=2
C
C     ***** MODE=1 IF ON INTERCOM; OTHERWISE, MODE=2. *****
C
      PRINT'(/T2,2A,I1,A/)','WELCOME TO OPSACT. INDICATE INTERCOM ',
     :                  'MODE BY ENTERING A "',1,'"'
      READ*, MODE
```

```
              PRINT'(/T2,2A,//16(T2,A/))',
     :           'IN ORDER THAT THE MOTOR NOISE BE PROPERLY AUGMENTED INTO ',
     :           'THE "W" MATRIX, THE "A" MATRIX MUST BE ARRANGED AS FOLLOWS:',
     :           '    ***         .        .                   ****',
     :           '    *           .        .                      *',
     :           '    * BASIC     .        .                      *',
     :           '    * SYSTEM    .        .                      *',
     :           '    *           .        .                      *',
     :           '    *...........         .                      *',
     :           '    *                    .                      *',
     :           '    * AUGMENTING         .                      *',
     :           '    * CONTROL INPUTS     .                      *',
     :           '    *                    .                      *',
     :           '    *.....................                      *',
     :           '    *                                           *',
     :           '    * AUGMENTING                                *',
     :           '    * CONTROL SYSTEM                            *',
     :           '    *                                           *',
     :           '    ***                                      ****'
C
C        ***** START *****
C
1        JABORT=0
         IF (MODE.EQ.1) THEN
            PRINT'(/T2,A)','ENTER RUN/CONFIGURATION NUMBER (INTEGER)'
            PRINT'(/T2,2A/T2,2A)','ENTER SYSTEM ORDER (N), CONTROL ',
     :            'ORDER (M) ','CONTROL SYSTEM ORDER (MM), ',
     :            'AND OUTPUT MEASUREMENT ORDER (YN).'
            PRINT'(T2,A/4(T6,A,I1,A/))','ENTER KTRAJ AS FOLLOWS:',
     :         'KTRAJ = ',0,' IF NO PLOT IS DESIRED.',
     :         'KTRAJ = ',1,' IF ONLY A PLOT IS DESIRED.',
     :         'KTRAJ = ',2,' IF ONLY A LISTING IS DESIRED.',
     :         'KTRAJ = ',3,' IF BOTH PLOT AND LISTING ARE DESIRED.'
            PRINT'(/T2,4A,I1,A)','FOR LISTING OF EIGENVALUES AND ',
     :                        'EIGENVECTORS OF THE CONTROLLABILITY/',
     :                        'OBSERVABILITY GRAMMIAN FROM MRIC, ',
     :                        'SET "EIG" = "',1,'".'
         ENDIF
         READ*,RUN,N,M,MM,YN,KTRAJ,EIG
         PRINT'(//T2,A,I2//)','RUN/CONFIGURATION NUMBER IS: ', RUN
         NDIM=12
         NDIM1=NDIM+1
         KOUT=6
         KPU=6
         KIN=5
         JJ=0
2        IF (CHANGE.EQ.1) THEN
            IF (MODE.EQ.1) THEN
               IF (JJ.EQ.0) THEN
                  PRINT*,'IF YOU WISH TO CHANGE ANY OF THE FOLLOWING ',
     :                 'INPUTS, ENTER THE NUMBER IN PARENTHESES BESIDE ',
     :                 'THE INPUT. ENTER A ZERO IF NO MORE CHANGES ARE ',
     :                 'DESIRED.'
                  PRINT'(/12(T2,A,I2,A))',
     :              'A MATRIX.....(',1,')',
     :              'B MATRIX.....(',2,')',
     :              'C MATRIX.....(',3,')',
```

```
          :              'Q MATRIX.....(',4,')',
          :              'R MATRIX.....(',5,')',
          :              'V MATRIX.....(',6,')',
          :              'W MATRIX.....(',7,')',
          :              'INITIAL TIME.(',8,')',
          :              'FINAL TIME...(',9,')',
          :              'TINCR........(',10,')',
          :              'X(INITIAL)...(',11,')',
          :              'U(INITIAL)...(',12,')',
          :              'THRESHOLD....(',13,')',
          :              'LAST ENTRY...(',0,')'
                       GO TO 6
                    ENDIF
                    PRINT'(/T2,A/)','ENTER NUMBER FOR NEXT DESIRED CHANGE.'
                 ENDIF
      6          READ*, IRVING
                 JJ=1
                 IF ((IRVING.LT.1.OR.IRVING.GT.13) THEN
                    GO TO 135
                 ENDIF
                 GO TO (10,20,30,40,50,60,70,80,90,100,110,120,130), IRVING
              ENDIF
      C
      10      IF (MODE.EQ.1) THEN
                 PRINT*, 'ENTER SYSTEM COEFFICIENT MATRIX (A). ENTER ROW, ',
          :               'COLUMN, AND VALUE; EACH SEPARATED BY A COMMA.'
              ENDIF
              HANDLE=' "A" '
              CALL INPMTX(HANDLE,A,N,N,A,CHANGE,MODE)
              IF (CHANGE.EQ.1) THEN
                 GO TO 2
              ENDIF
      20      IF (MODE.EQ.1) THEN
                 PRINT*, 'ENTER CONTROL INPUT MATRIX (B)'
              ENDIF
              HANDLE=' "B" '
              CALL INPMTX(HANDLE,B,N,M,B,CHANGE,MODE)
              IF (CHANGE.EQ.1) THEN
                 GO TO 2
              ENDIF
      30      IF (MODE.EQ.1) THEN
                 PRINT*, 'ENTER OUTPUT MEASUREMENT MATRIX (C)'
              ENDIF
              HANDLE=' "C" '
              CALL INPMTX(HANDLE,C,YN,N,C,CHANGE,MODE)
              IF (CHANGE.EQ.1) THEN
                 GO TO 2
              ENDIF
      40      IF (MODE.EQ.1) THEN
                 PRINT*,'ENTER STATE WEIGHTING MATRIX (Q)'
              ENDIF
              HANDLE=' "Q" '
              CALL INPMTX(HANDLE,Q,N,N,Q,CHANGE,MODE)
              IF (CHANGE.EQ.1) THEN
                 GO TO 2
              ENDIF
      50      IF (MODE.EQ.1) THEN
```

```
              PRINT*,'ENTER CONTROL WEIGHTING MATRIX (R)'
           ENDIF
           HANDLE='   "R"   '
           CALL INPMTX(HANDLE,R,M,M,R,CHANGE,MODE)
           IF (CHANGE.EQ.1) THEN
              GO TO 2
           ENDIF
   60      IF (MODE.EQ.1) THEN
              PRINT*,'ENTER OBSERVATION NOISE COVARIANCE MATRIX (V)'
           ENDIF
           HANDLE='   "V"   '
           CALL INPMTX(HANDLE,V,YN,YN,V,CHANGE,MODE)
           IF (CHANGE.EQ.1) THEN
              GO TO 2
           ENDIF
   70      IF (MODE.EQ.1) THEN
              PRINT*,'ENTER EXTERNAL DISTURBANCE COVARIANCE MATRIX (W)'
           ENDIF
           HANDLE='   "W"   '
           CALL INPMTX(HANDLE,W,N,N,W,CHANGE,MODE)
           IF (CHANGE.EQ.1) THEN
              GO TO 2
           ENDIF
   80      IF (MODE.EQ.1) THEN
              PRINT*, 'ENTER INITIAL TIME (TI) AS A REAL NUMBER'
           ENDIF
           READ*, TI
           PRINT'(/T2,A,F6.2/)', 'THE INITIAL TIME IS: ', TI
           IF (CHANGE.EQ.1) THEN
              GO TO 2
           ENDIF
   90      IF (MODE.EQ.1) THEN
              PRINT*,'ENTER FINAL TIME (TF) AS A REAL NUMBER'
           ENDIF
           READ*, TF
           PRINT'(/T2,A,F6.2/)','THE FINAL TIME IS: ',TF
           IF (CHANGE.EQ.1) THEN
              GO TO 2
           ENDIF
  100      IF (MODE.EQ.1) THEN
              PRINT*,'ENTER THE INTEGRATION TIME INCREMENT (TINCR)',
          :' AS A REAL NUMBER'
           ENDIF
           READ*, TINCR
           PRINT'(/T2,A,F6.3/)','THE TIME INCREMENT IS: ',TINCR
           IF (CHANGE.EQ.1) THEN
              GO TO 2
           ENDIF
  110      IF (MODE.EQ.1) THEN
              PRINT*,'ENTER INITIAL STATE VALUES'
              PRINT'(T2,A,I2,A,A/)','ENTER AS FOLLOWS: ROW, ',1,' VALUE;',
          :                          ' EACH SEPARATED BY A COMMA.'
           ENDIF
           HANDLE='   XI    '
           CALL INPMTX(HANDLE,XI,N,1,XI,CHANGE,MODE)
           IF (CHANGE.EQ.1) THEN
              GO TO 2
```

```
                ENDIF
          120   IF (MODE.EQ.1) THEN
                   PRINT*, 'ENTER INITIAL CONTROL INPUT VALUES'
                   PRINT'(T2,A,I2,A,A/)','ENTER AS FOLLOWS: ROW,',1,' VALUE;',
               :                           ' EACH SEPARATED BY COMMAS.'
                ENDIF
                HANDLE='  UI   '
                CALL INPMTX(HANDLE,UI,M,1,UI,CHANGE,MODE)
                IF (CHANGE.EQ.1) THEN
                   GO TO 2
                ENDIF
          130   IF (MODE.EQ.1) THEN
                   PRINT*, 'ENTER INSTRUMENT THRESHOLDS.'
                   PRINT'(T2,A,I2,A,A/)','ENTER AS FOLLOWS: ROW,',1,' VALUE;',
               :                           ' EACH SEPARATED BY COMMAS.'
                ENDIF
                HANDLE='THRSHLD'
                CALL INPMTX(HANDLE,THRSHD,YN,1,THRSHD,CHANGE,MODE)
                IF (CHANGE.EQ.1) THEN
                   GO TO 2
                ENDIF
          C
          C
          C     ***** CALCULATION OF OPTIMAL FEEDBACK GAIN MATRIX (L) *****
          C
          135   CALL TRANSP(B,BTRANS,N,M)
                CALL INPMTX(HANDLE,R,M,M,RESULT,3,MODE)
                CALL GMINV(M,M,RESULT,RINV,MR,0)
                IF (MR.NE.M) THEN
                   PRINT'(/T2,A/)', 'R NOT FULL RANK.'
                   IF (MODE.EQ.1) THEN
                      GO TO 1
                   ENDIF
                   JABORT=1
                   GO TO 250
                ENDIF
                CALL MMUL(RINV,BTRANS,M,M,N,RESULT)
                CALL MMUL(B,RESULT,N,M,N,RESULT2)
                IER=0
                CALL MRIC(N,A,RESULT2,Q,K,Z,IER)
                IF (IER.EQ.1) THEN
                   IF (MODE.EQ.1) THEN
                      GO TO 1
                   ENDIF
                   JABORT=1
                   GO TO 250
                ENDIF
                PRINT'(/T2,A/)', 'THE K MATRIX IS:'
                DO 200 I=1,N
                   PRINT'(10(3X,E10.4))',(K(I,J),J=1,N)
          200   CONTINUE
                CALL MMUL(RESULT,K,M,N,N,L)
                   PRINT'(10(3X,E10.4))',(L(I,J),J=1,N)
          201   CONTINUE
          C
          C     ***** DETERMINING NEUROMUSCULAR LAGS, TAU(N) *****
          C
```

```
      II=1
      DO 203 I=(N-MM)-(M-1),(N-MM)
         DO 202 J=(N-MM)-(M-1),(N-MM)
            TAUN(I,J)=1/L(II,J)
202      CONTINUE
         II=II+1
203   CONTINUE
C
C     ***** FORMING A1 AND B1 *****
C
      II=1
      DO 206 I=(N-MM)-(M-1),(N-MM)
         DO 205 J=(N-MM)-(M-1),(N-MM)
            A(I,J)=-1/TAUN(I,J)
            B(I,II)=1/TAUN(I,J)
            II=II+1
205      CONTINUE
         II=1
206   CONTINUE
C
      PRINT'(//T2,A)','A (ONE)'
      DO 207 I=1,N
         PRINT'(10(3X,E10.4))',(A(I,J),J=1,N)
207   CONTINUE
      PRINT'(//T2,A)','B (ONE)'
      DO 208 I=1,N
         PRINT'(10(3X,E10.4))',(B(I,J),J=1,M)
208   CONTINUE
C
C     ***** FORMING LO *****
C
      DO 210 I=1,M
         DO 209 J=(N-MM)-(M-1),(N-MM)
            L(I,J)=0.
209      CONTINUE
210   CONTINUE
      PRINT'(/T2,A)',' THE L (ZERO) MATRIX'
      DO 211 I=1,M
         PRINT'(10(3X,E10.4))',(L(I,J),J=1,N)
211   CONTINUE
C
C     ***** CALCULATION OF OPEN-LOOP EIGENVALUES *****
C
      CALL EIGEN(N,A,Z4,Z2,DUM,0)
      PRINT'(/T2,9X,A//14X,A,7X,A/)','OPEN LOOP EIGENVALUES',
     :                                'REAL','IMAGINARY'
      DO 219 I=1,N
         PRINT'(10X,2(E12.5,2X))',Z4(I),Z2(I)
219   CONTINUE
C
C     ***** CALCULATION OF CLOSED/LOOP SYSTEM EIGENVALUES ****
C
      CALL MMUL(B,L,N,M,N,RESULT)
      CALL MSUB(A,RESULT,N,N,Z1)
C
C     ***** NOTE: Z1 = A(ONE)-B(ONE)*L(ZERO) *****
C
```

```
              CALL EIGEN(N,Z1,Z4,Z2,DUM,0)
              PRINT'(/T2,9X,A//14X,A,7X,A/)','CLOSED LOOP EIGENVALUES',
     :                                       'REAL','IMAGINARY'
              DO 220 I=1,N
                 PRINT'(10X,2(E12.5,2X))',Z4(I),Z2(I)
        220   CONTINUE
        C
        C     ***** CALCULATION OF PHI, DREV, AND GAMMA *****
        C
              TAU=TINCR*8
              CALL MEXP(N,A,TINCR,PHI)
              CALL MEXP(N,A,TAU,DREV)
              CALL INTEAT(N,A,RESULT,TINCR)
              CALL MMUL(RESULT,B,N,N,M,GAMMA)
        C
        C     ***** CALCULATION OF THE ESTIMATOR GAIN MATRIX (S) *****
        C
              CALL TRANSP(A,ATRANS,N,N)
              CALL TRANSP(C,CTRANS,YN,N)
        221   CALL INPMTX(HANDLE,V,YN,YN,RESULT,3,MODE)
              CALL GMINV(YN,YN,RESULT,VINV,MR,0)
              IF (MR.NE.YN) THEN
                 PRINT'(/T2,A/)', 'V NOT FULL RANK.'
                 IF (MODE.EQ.1) THEN
                    GO TO 1
                 ENDIF
                 JABORT=1
                 GO TO 250
              ENDIF
              CALL MMUL(CTRANS,VINV,N,YN,YN,RESULT)
              CALL MMUL(RESULT,C,N,YN,N,RESULT2)
              CALL MRIC(N,ATRANS,RESULT2,W,P,Z1,IER)
              IF (IER.EQ.1) THEN
                 IF (MODE.EQ.1) THEN
                    GO TO 1
                 ENDIF
                 JABORT=1
                 GO TO 250
              ENDIF
              CALL MMUL(P,RESULT,N,N,YN,S)
        C
        C     ***** CALCULATION OF PSI *****
        C
              CALL INTEAT(N,A,RESULT,TINCR)
              CALL MMUL(RESULT,S,N,N,YN,PSI)
        C
        C     ***** CALCULATION OF THE COVARIENCE MATRICES *****
        C     ***** OF THE STATES AND THE MEASUREMENTS.    *****
        C
              CALL MMUL(DREV,P,N,N,N,RESULT)
              CALL MEXP(N,ATRANS,TAU,Z1)
              CALL MMUL(RESULT,Z1,N,N,N,RESULT3)
              CALL INPMTX(HANDLE,W,N,N,RESULT,3,MODE)
              CALL INTEG(N,A,RESULT,RESULT2,TAU)
              CALL MADD(RESULT3,RESULT2,N,N,RESULT)
              CALL MMUL(DREV,P,N,N,N,RESULT2)
              CALL MMUL(RESULT2,CTRANS,N,N,YN,RESULT3)
```

```fortran
      CALL MMUL(RESULT3,VINV,N,YN,YN,RESULT2)
      CALL MMUL(RESULT2,C,N,YN,N,RESULT3)
      CALL MMUL(RESULT3,P,N,N,N,RESULT2)
      CALL MMUL(RESULT2,Z1,N,N,N,RESULT3)
      TOL=.001
      CALL TRANSP(Z,RESULT2,N,N)
      CALL MLINEQ(N,RESULT2,RESULT3,Z1,TOL)
      CALL MADD(RESULT,Z1,N,N,RESULT3)
      CALL MMUL(C,RESULT3,YN,N,N,RESULT)
      CALL MMUL(RESULT,CTRANS,YN,N,YN,RESULT2)
C
C     ***** NOTE:  Z = A(BAR) = A(ZERO)-B(ZERO)*L(ZERO)         *****
C     *****        RESULT2 = DISPLAY VARIABLE COVARIANCE MATRIX *****
C     *****        RESULT3 = STATE COVARIANCE MATRIX            *****
C
C
C     ***** COMPARING GUESSED NOISES TO VARIENCES *****
C
      GUESS=0
      DO 223 I=1,YN
         SUM1=SQRT(RESULT2(I,I)*2)
         SUM1=THRSHD(I)/SUM1
         SUM2=ERFC(SUM1)
         SUM1=SUM2**2
         RESULT(I,I)=.0314159*(RESULT2(I,I)/SUM1)
         SUM1=ABS((V(I,I)-RESULT(I,I))/RESULT(I,I))
         IF (SUM1.LT..05) THEN
            GO TO 223
         ENDIF
         V(I,I)=RESULT(I,I)
         GUESS=1
  223 CONTINUE
      DO 224 I=(N-MM)-(M-1),(N-MM)
         Z1(I,I)=.0094248*RESULT3(I,I)
         W(I,I)=W(I,I)*(TAUN(I,I)**2)
         SUM1=ABS((W(I,I)-Z1(I,I))/Z1(I,I))
         IF (SUM1.LT..05) THEN
            W(I,I)=W(I,I)/(TAUN(I,I)**2)
            GO TO 224
         ENDIF
         W(I,I)=Z1(I,I)
         W(I,I)=W(I,I)/(TAUN(I,I)**2)
         GUESS=1
  224 CONTINUE
C
C     ***** NOTE: RESULT = (.01 PI*DISPLAY COVARIANCE)/(F(HAT))**2 *****
C     *****       Z1 = .003 PI*STATE COVARIANCE                    *****
C
      IF (GUESS.EQ.1) THEN
         GO TO 221
      ENDIF
C
C
      PRINT*,'STATE COVARIANCE MATRIX'
      DO 239 I=1,N
      PRINT'(10(3X,E10.4))',(RESULT3(I,J),J=1,N)
  239 CONTINUE
```

```
  240   PRINT'(/T2,A/)', 'THE P MATRIX IS:'
        DO 241 I=1,N
           PRINT'(10(3X,E10.4))',(P(I,J),J=1,N)
  241   CONTINUE
        PRINT '(/T2,A/)', ' THE KALMAN FILTER GAIN MATRIX (S) IS: '
        DO 242 I=1,N
           PRINT'(10(3X,E10.4))',(S(I,J),J=1,YN)
  242   CONTINUE
  C
  C     ***** CALCULATION OF CLOSED-LOOP ESTIMATOR EIGENVALUES *****
  C
        CALL MMUL(S,C,N,YN,N,Z)
        CALL MSUB(A,Z,N,N,Z1)
  C
  C     ***** NOTE: Z1 = A(ONE)-S*C *****
  C
        CALL EIGEN(N,Z1,Z4,Z2,DUM,O)
        PRINT'(/T2,9X,A//14X,A,7X,A/)',
       :     'CLOSED-LOOP ESTIMATOR EIGENVALUES','REAL','IMAGINARY'
        DO 243 I=1,N
           PRINT'(10X,2(E12.5,2X))',Z4(I),Z2(I)
  243   CONTINUE
  C
  C     ***** REFORMING AO AND BO FOR NEXT RUN *****
  C
        III=1
        DO 246 I=(N-MM)-(M-1),(N-MM)
           DO 244 II=1,M
              B(I,II)=0.
  244      CONTINUE
           DO 245 J=(N-MM)-(M-1),(N-MM)
              A(I,J)=0.
  245      CONTINUE
           B(I,III)=1.
           III=III+1
  246   CONTINUE
  C
  C     ***** THEN RETURN TO START FOR THE NEXT PROBLEM.          *****
  C     ***** ON INTERCOM, READ REMAINING CARDS FOR THIS PROBLEM. *****
  C     ***** IF ON INTERCOM AND JABORT=1, RETURN TO START. IF NOT *****
  C
  250   IF (KTRAJ.EQ.2) THEN
           GO TO 260
        ELSEIF (KTRAJ.EQ.0) THEN
           GO TO 290
        ENDIF
  C
  C     ***** PLOT/LIST INFORMATION *****
  C
        IF (MODE.EQ.1) THEN
           PRINT'(/T2,A/T2,A,I2,A/)', 'ENTER TITLE OF PLOT,',
       :        '(',30,' CHARACTERS MAX).'
        ENDIF
        READ (5,'(3A10)') TITLE
  260   IF (MODE.EQ.1) THEN
           PRINT'(/T2,A/T8,A/T8,2A/)',
       :        'ENTER THE FOLLOWING INFORMATION FOR PLOT/LIST:',
```

```
                  :           'PLOT TIME INCREMENT (DT) AS A REAL NUMBER,',
                  :           'NUMBER OF OUTPUT/STATE TIME RESPONSES DESIRED ',
                  :           'FOR EACH PLOT, (MAX IS FOUR).'
                   ENDIF
                   READ*, DT,NOUT
                   IF (NOUT.GT.4) THEN
                      PRINT'(/T2,A/)','WAKE UP, EIGHT-BALL. I SAID THE MAX IS FOUR.'
                      IF (MODE.EQ.1) THEN
                         GO TO 260
                      ENDIF
                      GO TO 265
                   ENDIF
                   IF (MODE.EQ.1) THEN
                      PRINT'(/T2,2A,I1,A/T2,A,I1,2A/T2,2A,I1,A/T2,2A)',
                  :        'FOR EACH STATE TIME RESPONSE DESIRED, ENTER ',
                  :        'A ',1,' X N MATRIX (COUT) OF ZEROS WITH',
                  :        'A "',1,'" PLACED IN THE POSITION CORRESPONDING ',
                  :        'TO THE DESIRED STATE.',
                  :        'FOR EACH OUTPUT TIME RESPONSE DESIRED ENTER ',
                  :        'A ',1,' X N MATRIX (COUT) CONSISTING OF THE ',
                  :        'ROW OF THE C MATRIX CORRESPONDING TO THE ',
                  :        'DESIRED OUTPUT (Y).'
                   ENDIF
         265       DO 266 I=1,NOUT
                      READ (5,*) (COUT(I,J),J=1,N)
         266       CONTINUE
                   PRINT'(/T2,A/)', 'THE PLOT/LIST OUTPUT MATRIX (COUT) IS:'
                   DO 267 I=1,NOUT
                      PRINT'(10(3X,E10.4))',(COUT(I,J),J=1,N)
         267       CONTINUE
                   IF (KTRAJ.EQ.2) THEN
                      GO TO 280
                   ENDIF
                   IF (MODE.EQ.1) THEN
                      PRINT'(/T2,A/)', 'ENTER LABELS FOR EACH OUTPUT.'
                   ENDIF
         268       DO 275 I=1,NOUT
                      GO TO (270,271,272,273),I
         270                READ (5,'(2A10)') Y1TITLE(1),Y1TITLE(2)
                            GO TO 275
         271                READ (5,'(2A10)') Y2TITLE(1), Y2TITLE(2)
                            GO TO 275
         272                READ (5,'(2A10)') Y3TITLE(1), Y3TITLE(2)
                            GO TO 275
         273                READ (5,'(2A10)') Y4TITLE(1), Y4TITLE(2)
         275       CONTINUE
         C
         C
         280       NPOINTS=(TF-T0)/DT
                   NLEN=300
                   IF (NPOINTS.GT.298.) THEN
                      NPOINTS=298.
                   ENDIF
                   II=0
                   COUNT=0
                   LIMIT=DT/TINCR
         290       IF (JABORT.EQ.1) THEN
```

```
            GO TO 402
         ENDIF
C
C        ******************************************************************
C        *     DEFINING OPTIMAL PILOT VARIABLES                           *
C        *       UOPT:     PRESENT OPTIMAL INPUT                          *
C        *       UOLD:     PAST OPTIMAL INPUT                             *
C        *       UDELAY:   INPUT DELAYING MATRIX                          *
C        *       DUOPT:    DELAYED OPTIMAL INPUT                          *
C        *       XNEW:     PRESENT STATES                                 *
C        *       XOLD:     PAST STATES                                    *
C        *       YNEW:     PRESENT DISPLAY VARIABLES                      *
C        *       YDELAY:   DISPLAY VARIABLES DELAYING MATRIX              *
C        *       Y:        DELAYED DISPLAY VARIABLES                      *
C        *       YEST:     ESTIMATOR DISPLAY VARIABLE ERROR               *
C        *       DXHATN:   PRESENT ESTIMATED DELAYED STATES               *
C        *       DXHATO:   PAST ESTIMATED DELAYED STATES                  *
C        *       DXHATN:   PRESENT PREDICTED STATES DUE TO OPTIMAL        *
C        *                 INPUTS (PREDICTED OPTIMAL STATES)              *
C        *       DXHATO:   PAST PREDICTED OPTIMAL STATES                  *
C        *       XDELAY:   PREDICTED OPTIMAL STATES DELAYING MATRIX       *
C        *       DDXHAT:   DELAYED PREDICTED OPTIMAL STATES               *
C        *       DEE:      DELAYED STATE RECONSTRUCTION ERROR             *
C        *       EE:       STATE RECONSTRUCTION ERROR                     *
C        *       XHAT:     RECONSTRUCTED STATES                           *
C        *       PFM:      PERFORMANCE INDEX (ONE ITERATION)              *
C        *       PFMT:     TOTAL PERFORMANCE INDEX                        *
C        ******************************************************************
C
C
C        ***** INITIALING OPTIMAL PILOT VARIABLES *****
C
         PFM=0.0
         PFMT=0.0
         X3=0.
         U3=0.
         DO 292 I=1,NDIM
            DO 291 J=1,NDIM
               X2(I,J)=0.
               U2(I,J)=0.
               YDELAY(I,J)=0.
               XDELAY(I,J)=0.
               UDELAY(I,J)=0.
  291       CONTINUE
  292    CONTINUE
         DO 297 I=1,N
            YEST(I)=0.
            UOLD(I)=0.
            UOPT(I)=0.
            XNEW(I)=0.
            DXHATN(I)=0.
            DXHATO(I)=0.
            DEE(I)=0.
            EE(I)=0.
            XHAT(I)=0.
            DXHATN(I)=0.
            XOLD(I)=0.
```

```
                 DXHATO(I)=0.
                 DUOPT(I)=0.
                 DDXHAT(I)=0.
                 YNEW(I)=0.
                 Y(I)=0.
      297     CONTINUE
              TOLD=TI
              CALL INPMTX(HANDLE,XI,N,1,XOLD,3,MODE)
              CALL INPMTX(HANDLE,XI,N,1,DXHATO,3,MODE)
              CALL INPMTX(HANDLE,UI,M,1,UOLD,3,MODE)
      C
      C       ***** THE OPTIMAL PILOT LOOP *****
      C
      300     TNEW=TOLD+TINCR
              IF (TNEW.GE.TF) THEN
                 GO TO 400
              ENDIF
      C
      C       ***** CALCULATION OF THE OPTIMAL INPUT *****
      C
              CALL SMMULT(L,-1.,M,N,RESULT)
              CALL MMUL(RESULT,XHAT,M,N,1,UOPT)
      C
      C       ***** DELAYING THE OPTIMAL INPUT BY TAU SECONDS *****
      C
              DO 311 J=1,8
                 DO 310 I=1,M
                    UDELAY(I,J)=UDELAY(I,J+1)
      310        CONTINUE
      311     CONTINUE
              DO 312 I=1,M
                 UDELAY(I,9)=UOPT(I)
      312     CONTINUE
              DO 313 I=1,M
                 DUOPT(I)=UDELAY(I,1)
      313     CONTINUE
      C
      C       ***** CALCULATION OF SYSTEM STATES *****
      C
              CALL MMUL(PHI,XOLD,N,N,1,RESULT)
              CALL MMUL(GAMMA,UOPT,N,M,1,RESULT2)
              CALL MADD(RESULT,RESULT2,N,1,XNEW)
      C
      C       ***** CALCULATION AND DELAY OF DISPLAYED VARIABLES *****
      C
              DO 321 J=1,8
                 DO 320 I=1,YN
                    YDELAY(I,J)=YDELAY(I,J+1)
      320        CONTINUE
      321     CONTINUE
              CALL MMUL(C,XNEW,YN,N,1,YNEW)
              DO 322 I=1,YN
                 YDELAY(I,9)=YNEW(I)
      322     CONTINUE
              DO 323 I=1,YN
                 Y(I)=YDELAY(I,1)
      323     CONTINUE
```

```
      C
      C     ***** CALCULATION OF ESTIMATED STATES *****
      C
            CALL MMUL(C,DXHATN,YN,N,1,RESULT)
            DO 330 I=1,YN
               YEST(I)=Y(I)-RESULT(I,1)
      330   CONTINUE
      C
            CALL MMUL(PHI,DXHATO,N,N,1,RESULT)
            CALL MMUL(GAMMA,DUOPT,N,M,1,RESULT2)
            CALL MADD(RESULT,RESULT2,N,1,RESULT)
            CALL MMUL(PSI,YEST,N,YN,1,RESULT2)
            CALL MADD(RESULT,RESULT2,N,1,DXHATN)
      C
      C     ***** CALCULATION OF PREDICTED OPTIMAL STATES *****
      C
            CALL MMUL(PHI,DXHATO,N,N,1,RESULT)
            CALL MMUL(GAMMA,UOPT,N,M,1,RESULT2)
            CALL MADD(RESULT,RESULT2,N,1,DXHATN)
      C
      C     ***** DELAYING PREDICTED OPTIMAL STATES *****
      C
            DO 341 J=1,8
               DO 340 I=1,N
                  XDELAY(I,J)=XDELAY(I,J+1)
      340      CONTINUE
      341   CONTINUE
            DO 342 I=1,N
               XDELAY(I,9)=DXHATN(I)
      342   CONTINUE
            DO 343 I=1,N
               DDXHAT(I)=XDELAY(I,1)
      343   CONTINUE
      C
      C     ***** CALCULATION OF RECONSTRUCTED SYSTEM STATES (XHAT) *****
      C
            CALL MSUB(DXHATN,DDXHAT,N,1,DEE)
            CALL MMUL(DREV,DEE,N,N,1,EE)
            CALL MADD(EE,DXHATN,N,1,XHAT)
      C
      C     ***** UPDATING VARIABLES FOR NEXT ITERATION *****
      C
            TOLD=TNEW
            DO 350 I=1,N
               XOLD(I)=XNEW(I)
               DXHATO(I)=DXHATN(I)
               DXHATO(I)=DXHATN(I)
      350   CONTINUE
            DO 351 I=1,M
               UOLD(I)=UOPT(I)
      351   CONTINUE
      C
      C
      C     ***** CALCULATION OF THE PERFORMANCE MEASURE *****
      C
            CALL TRANSP(XOLD,RESULT,N,1)
            CALL MMUL(XOLD,RESULT,N,1,N,X2)
```

```
                CALL DMMUL(X2,Q,N,X3)
                CALL TRANSP(UOLD,RESULT2,M,1)
                CALL MMUL(UOLD,RESULT2,M,1,M,U2)
                CALL DMMUL(U2,R,M,U3)
                PFM=X3+U3
                PFMT=PFMT+PFM
          C
                IF (KTRAJ.EQ.0) THEN
                   GO TO 300
                ENDIF
          C
          C
          C     ***** STORAGE OF INFORMATION FOR PLOT/LISTING *****
          C
                COUNT=COUNT+1
                IF (COUNT.EQ.1) THEN
                   II=II+1
                   TP(II)=TOLD
                   CALL MMUL(COUT,XOLD,NOUT,N,1,OUTPUT)
                   INDEX=II
                DO 360 I=1,NOUT
                      YA(INDEX)=OUTPUT(I)
                      INDEX=INDEX+NLEN
          360   CONTINUE
                ENDIF
                IF (COUNT.GE.LIMIT) THEN
                   COUNT=0
                ENDIF
          C
          C     ***** END OF OPTIMAL PILOT LOOP *****
          C
                GO TO 300
          C
          C     ***** PLOT/LISTING *****
          C
          400   PRINT'(/T2,A,E20.6/)','THE VALUE OF THE PERFORMANCE INDEX IS: ',
               :      PFMT
          402   IF (KTRAJ.EQ.0) THEN
                   GO TO 420
                ENDIF
                IF (JABORT.EQ.1) THEN
                   GO TO 410
                ENDIF
                IF (KTRAJ.EQ.1.OR.KTRAJ.EQ.3) THEN
                   CALL DRAWZ(TP,YA,YA(301),YA(601),YA(901),NOUT,NOUT,NPOINTS,0,1)
                ENDIF
                IF (KTRAJ.EQ.2.OR.KTRAJ.EQ.3) THEN
                   IF (MODE.EQ.1) THEN
                      IG=(NPOINTS+2)/3
                   ELSE
                      IG=(NPOINTS+4)/5
                   ENDIF
                   DO 404 LA=1,NOUT
                      IF (MODE.EQ.1) THEN
                         PRINT'(/T4,3(A,10X,A,I1,9X)/)','T','Y',LA
                      ELSE
                         PRINT'(/T4,5(A,10X,A,I1,9X)/)','T','Y',LA
```

```
                    ENDIF
                 INC=(LA-1)*NLEN
                 DO 403 IE=1,IG
                    PRINT'(T2,5(F8.3,F12.4,6X))',
     :                    (TP(ID),YA(ID+INC),ID=IE,NPOINTS,IG)
      403           CONTINUE
      404        CONTINUE
              ENDIF
      C
      C       ***** IF MORE PLOTS/LISTINGS FOR THIS PROBLEM *****
      C       ***** ARE DESIRED, SET "MORE"=2.              *****
      C
      410     IF (MODE.EQ.1) THEN
                 PRINT'(/T2,A/T2,A,I1/)', 'IF MORE PLOTS/LISTINGS FOR',
     :                 'THIS PROBLEM ARE DESIRED, ENTER A ',2
              ENDIF
              READ*, MORE
              IF (JABORT.EQ.1) THEN
                 GO TO 420
              ENDIF
              IF (MORE.EQ.2) THEN
                 GO TO 250
              ENDIF
      C
      C       ***** IF CHANGES TO THE SYSTEM/PROBLEM ARE TO BE MADE AND *****
      C       ***** A NEW PROBLEM CREATED, SET "CHANGE"=1.              *****
      C
      420     IF (MODE.EQ.1) THEN
                 PRINT'(/T2,A/T2,A,I1/)', 'IF CHANGES TO THE SYSTEM/PROBLEM',
     :                 'ARE TO BE MADE AND A NEW PROBLEM CREATED, ENTER A ', 1
              ENDIF
              READ*,CHANGE
              IF (CHANGE.EQ.1) THEN
                 GO TO 1
              ENDIF
      C
      500     END
```

SUBROUTINE INPMTX

```
      SUBROUTINE INPMTX(NAME,OLDMAT,NR,NC,MATRIX,CHANGE,MODE)
C
C        SUBROUTINE INPMTX IS DESIGNED TO INPUT OR CHANGE THE VALUES
C     OF A 2-DIMENSIONAL MATRIX. THE VALUE OF "CHANGE" IS 1 IF THE
C     VALUES OF AN EXISTING MATRIX ARE TO BE CHANGED. A VALUE OF 2 IS
C     USED IF A NEW MATRIX IS TO BE ENTERED. A VALUE OF 3 IS USED IF
C     A NEW MATRIX IS TO BE SET EQUAL TO AN OLD MATRIX.
C
      REAL MATRIX(NDIM,1), OLDMAT(NDIM,1), NEWVAL(144)
      INTEGER I,J,CHANGE,NC,NR,MODE,ROW(144),COL(144)
      CHARACTER NAME*7
      COMMON/MAIN1/NDIM
C
      IF (CHANGE.EQ.1.OR.CHANGE.EQ.3) THEN
         GO TO 3
      ENDIF
      DO 2 I=1,NR
         DO 1 J=1,NC
            MATRIX(I,J)=0.0
 1       CONTINUE
 2    CONTINUE
      GO TO 6
 3    DO 5 I=1,NR
         DO 4 J=1,NC
            MATRIX(I,J)=OLDMAT(I,J)
 4       CONTINUE
 5    CONTINUE
      IF (CHANGE.EQ.3) THEN
         GO TO 10
      ENDIF
 6    READ (5,*) (ROW(I),COL(I),NEWVAL(I), I=1,NR*NC)
         DO 7 I=1,NR*NC
            IF (ROW(I).EQ.0) THEN
               GO TO 8
            ELSE
               MATRIX(ROW(I),COL(I))=NEWVAL(I)
            ENDIF
 7    CONTINUE
 8    PRINT '(/T2,3A/)', 'THE ', NAME, ' MATRIX IS:'
      DO 9 I=1,NR
         PRINT '(10(3X,E10.4))',(MATRIX(I,J),J=1,NC)
 9    CONTINUE
      IF(MODE.EQ.1) THEN
         PRINT'(T2,A,I2)','IF YOU WANT TO CHANGE A VALUE, ENTER A ',1
         PRINT '(T2,A,I2)','OTHERWISE, ENTER A ',2
         READ*, OOPS
         IF (OOPS.EQ.1) THEN
            PRINT*, 'ENTER CORRECTED ROW,COLUMN,AND VALUE',
            PRINT'(T2,A,I1,A/)', 'ENTER ',0,'/ AS YOUR LAST ENTRY'
            GO TO 6
         ELSE
            GO TO 10
         ENDIF
      ENDIF
 10   CONTINUE
      RETURN
      END
```

SUBROUTINE DMMUL

```
      SUBROUTINE DMMUL(A,B,N,SUM)
C
C        THIS SUBROUTINE MULTIPLIES THE DIAGONAL ELEMENTS
C     OF MATRIX A TO THE CORRESPONDING DIAGONAL ELEMENTS
C     OF MATRIX B, THEN SUMS THE VALUES AND RETURNS AS SCALAR SUM.
C
      REAL A(NDIM,N), B(NDIM,N), SUM
      INTEGER N
      COMMON/MAIN1/NDIM
      SUM=0
      DO 1 I=1,N
         SUM=SUM+A(I,I)*B(I,I)
    1 CONTINUE
      RETURN
      END
```

SUBROUTINE SMMULT

```
      SUBROUTINE SMMULT (MTX1,SCALAR,NR,NC,MTX2)
C
C        SUBROUTINE SMMULT PERFORMS THE OPERATION
C     MTX2=SCALAR*MTX1, WHERE NR IS THE NUMBER OF
C     ROWS OF MTX1 AND NC IS THE NUMBER OF COLUMNS.
C
C
      REAL MTX1(NDIM,1), MTX2(NDIM,1)
         INTEGER NR,NC
         REAL SCALAR, PROD
      COMMON/MAIN1/NDIM
         DO 2 I=1,NR
            DO 1 J=1,NC
               PROD=0.0
               PROD=MTX1(I,J)*SCALAR
               MTX2(I,J)=PROD
    1       CONTINUE
    2    CONTINUE
         RETURN
         END
```

SUBROUTINE TRANSP

```
      SUBROUTINE TRANSP (MTX1,MTX2,NR,NC)
C
C        SUBROUTINE TRANSP PERFORMS THE OPERATION
C     MTX2=MTX1 (TRANSPOSE), WHERE NR IS THE NUMBER OF
C     ROWS OF MTX1 AND NC IS THE NUMBER OF COLUMNS.
C
C
         REAL MTX1(NDIM,1), MTX2(NDIM,1)
         INTEGER NR,NC
      COMMON/MAIN1/NDIM
         DO 2 I=1,NR
            DO 1 J=1,NC
               MTX2(J,I)=MTX1(I,J)
    1       CONTINUE
    2    CONTINUE
         RETURN
         END
```

Kleinman Subroutines

SUBROUTINE GMINV

```
      SUBROUTINE GMINV(NR,NC,A,U,MR,MT)
      DIMENSION A(1),U(1),S(80)
      COMMON/MACHIL/ NDIM,NDIM1
      COMMON/INOU/KIN,KOUT
      TOL=1.E-12
      ADV=1.E-24
      MR=NC
      NRM1=NR-1
      TOL1=0.
      JJ=1
      DO 10 J=1,NC
      S(J)=DOT(NR,A(JJ),A(JJ))
      IF(S(J).GT.TOL1)TOL1=S(J)
   10 JJ=JJ+NDIM
      TOL1=ADV*TOL1
      ADV=TOL1
      JJ=1
      DO 100 J=1,NC
      FAC=S(J)
      JM1=J-1
      JRM=JJ+NRM1
      JCM=JJ+JM1
      DO 20 I=JJ,JCM
   20 U(I)=0.
      U(JCM)=1.0
      IF(J.EQ.1) GO TO 54
      KK=1
      DO 30 K=1,JM1
      IF(S(K).EQ.1.0) GO TO 30
      TEMP=-DOT(NR,A(JJ),A(KK))
      CALL VADD(K,TEMP,U(JJ),U(KK))
   30 KK=KK+NDIM
      DO 50 L=1,2
      KK=1
      DO 50 K=1,JM1
      IF(S(K).EQ.0.) GO TO 50
      TEMP=-DOT(NR,A(JJ),A(KK))
      CALL VADD(NR,TEMP,A(JJ),A(KK))
      CALL VADD(K,TEMP,U(JJ),U(KK))
   50 KK=KK+NDIM
      TOL1=TOL*FAC+ADV
      FAC=DOT(NR,A(JJ),A(JJ))
   54 IF(FAC.GT.TOL1) GO TO 70
      DO 55 I=JJ,JRM
   55 A(I)=0.
      S(J)=0.
      KK=1
      DO 65 K=1,JM1
      IF(S(K).EQ.0.) GO TO 65
      TEMP=-DOT(K,U(KK),U(JJ))
      CALL VADD(NR,TEMP,A(JJ),A(KK))
   65 KK=KK+NDIM
      FAC=DOT(J,U(JJ),U(JJ))
      MR=MR-1
      GO TO 75
   70 S(J)=1.0
      KK=1
```

```
      DO 65 JJ=I,II,NDIM
      C(J)=C(JJ)
   65 J=J+1
   70 ID=J
      DT1=C(J)
      DO 75 JJ=II,NN,NDIM
      C(J)=C(J)+DOT(N,F(II),X(JJ))
   75 J=J+1
      J=J-1
      DO 80 JJ=II,J
   80 X(JJ)=F(JJ)
      IF(ABS(C(ID)-DT1).LT.(ADV+TOL*ABS(C(ID)))) NEZ=NEZ+1
      I=I+1
      II=II+NDIM
      SIZE=SIZE+DT1
      IF(I.LE.N) GO TO 60
      IF(NEZ.EQ.N) GO TO 150
      IF(ABS(SIZE).GT.1.E13) GO TO 95
      CALL MMUL(X,X,N,N,N,F)
      IF(IT.EQ.ITT)GO TO 150
   90 CONTINUE
   95 WRITE(KOUT,100) IT
  100 FORMAT(33HOLIN SGN ALGORITHM NON-CONVERGENT,I3,10HITERATIONS)
  150 CALL EQUATE(X,C,N,N)
      RETURN
      END
```

FUNCTION XNORM

```
      FUNCTION XNORM(N,A)
C          COMPUTES AN APPROXIMATION TO NORM OF A-- NOT A BOUND
      DIMENSION A(1)
      COMMON/MAIN1/ NDIM,NDIM1
      NN=N*NDIM
      C1=0.
      TR=A(1)
      IF(N.EQ.1) GO TO 20
      I=2
      DO 10 II=NDIM1,NN,NDIM
      J=II
      DO 5 JJ=I,II,NDIM
      C1=C1+ABS(A(J)*A(JJ))
    5 J=J+1
      TR=TR+A(J)
   10 I=I+1
      TR=TR/FLOAT(N)
      DO 15 II=1,NN,NDIM1
   15 C1=C1+(A(II)-TR)**2
   20 XNORM=ABS(TR)+SQRT(C1)
      RETURN
      END
```

SUBROUTINE EGVEC

```
      SUBROUTINE EGVEC(N,A,RR,RI,U)
      DIMENSION A(1),RR(1),RI(1),U(1)
      COMMON/MAIN1/NDIM,MDIM
      GAMA=1.E-7
      T=0.
      NN=N*NDIM-NDIM
      DO 10 I=1,N
      DO 10 J=I,NN,NDIM
   10 T=T+ABS(A(J))
      TOL=T*GAMA
      MM=NN
C         BEGIN BACK SUBSTITUTION
      DO 180 KN=1,N
      NA=N-KN
      NE=1+NA
      P=RR(NE)
      B=RI(NE)
      KK=MM
      IF(B.NE.0.) GO TO 100
      A(MM)=1.
      IF(NA.LT.1) GO TO 190
      DO 60 KL=1,NA
      KK=KK-NDIM
      W=A(KK)-P
      J=MM-KL
      R=DOT3(KL,A(KK+NDIM),A(J+1))
      T=RI(NE-KL)
      IF(T)25,30,40
   25 Z=W
      S=R
      A(J)=0.
      GO TO 60
   30 IF(ABS(W).LE.TOL) W=TOL
      A(J)=-R/W
      GO TO 60
   40 X=A(KK+NDIM)
      Y=A(KK+1)
      Q=RR(NE-KL)-P
      A(J)=(X*S-Z*R)/(Q*Q+T*T)
      IF(ABS(X).LE.ABS(Z)) GO TO 50
      A(J+1)=(-R-W*A(J))/X
      GO TO 60
   50 A(J+1)=(-S-Y*A(J))/Z
   60 CONTINUE
      GO TO 180
  100 IF(B.GE.0.) GO TO 130
      IF(ABS(A(KK-NDIM)).LE.ABS(A(KK-1))) GO TO 110
      A(KK-NDIM)=-(A(KK)-P)/A(KK-NDIM)
      A(KK-1)=-B/A(KK-NDIM)
      GO TO 120
  110 AP=A(KK-NDIM)-P
      CALL CDIV(-A(KK-1),0.,AP,B,A(KK-NDIM),A(KK-1))
  120 A(KK-NDIM)=1.
      A(KK)=0.
      IF(NA.LT.2) GO TO 190
      KK=KK-NDIM
      DO 170 KL=2,NA
```

```
      J=MM-KL
      JP1=J+1
      KK=KK-NDIM
      W=A(KK)-R
      RA=DOT3(KL,A(KK+NDIM),A(JP1-NDIM))
      SA=DOT3(KL,A(KK+NDIM),A(JP1))
      T=R(MM-KL)
      IF(T)125,130,140
  125 Z=W
      R=RA
      S=SA
      A(J)=0.0
      A(J-NDIM)=0.0
      GO TO 170
  130 CALL CDIV(-RA,-SA,W,B,A(J-NDIM),A(J))
      GO TO 170
  140 X=A(KK+NDIM)
      Y=A(KK+1)
      VI=RR(MM-KL)-R
      VR=VI*VI+T*T-R*R
      VI=2.*VI*R
      IF(VR.GT.TOL.OR.VI.GT.TOL) GO TO 150
      VR=TOL*(ABS(W)+ABS(R)+ABS(X)+ABS(Y)+ABS(Z))

  150 AR=X*R-Z*RA+S*SA
      AI=X*S-Z*SA-R*RA
      CALL CDIV(AR,AI,VR,VI,A(J-NDIM),A(J))
      IF(ABS(X).LE.(ABS(Z)+ABS(B))) GO TO 160
      A(JP1-NDIM)=(-RA-W*A(J-NDIM)+B*A(J))/X
      A(JP1)=(-SA-W*A(J)-B*A(J-NDIM))/X
      GO TO 170
  160 AR=-R-Y*A(J-NDIM)
      AI=-S-Y*A(J)
      CALL CDIV(AR,AI,Z,B,A(JP1-NDIM),A(JP1))
  170 CONTINUE
  180 MM=MM-NDIM
C          MULTIPLY OUT BY XFORMATION AND SCALE
      DO 250 KN=1,N
      KI=1+NN-N
      J=1+N-KN
      T=RI(J)
      IF(T.GT.0.) GO TO 250
      I=1
      B=0.
      DO 230 KL=KI,NN
      Z=DOT3(J,Q(I),A(KI))
      B=B+Z*Z
      IF(T.EQ.0.)GO TO 220
      Y=DOT3(J,Q(I),A(KI-NDIM))
      B=B+Y*Y
      Q(KL-NDIM)=Y
  220 Q(KL)=Z
  230 I=I+1
      IF(B.LE.GAMA*GAMA) GOTO 250
      B=1./SQRT(B)
      DO 240 KL=KI,NN
      IF(T.LT.0.)Q(KL-NDIM)=B*Q(KL-NDIM)
  240 Q(KL)=B*Q(KL)
  250 NN=NN-NDIM
      RETURN
      END
```

SUBROUTINE INTEAT

```
      SUBROUTINE INTEAT(N,A,B,T)
C        B=INTEGRAL OF EXP(A*TAU) FROM 0 TO T
      DIMENSION A(1),B(1)
      COMMON/MAIN1/NDIM,NDIM1,X(1)
      COMMON/MAIN2/Y(1)
      COMMON/MAIN3/Z(1)
      IF (N.GT.1) GO TO 1
      B(1)=T
      IF (A(1).EQ.0.) RETURN
      B(1)=(EXP(T*A(1))-1.)/A(1)
      RETURN
    1 NN=N*NDIM
      NM1=N-1
      IND=0
      ANORM=XNORM(N,A)
      DT=T
    5 IF (ABS(DT).GT.1.) GO TO 10
      IF (ANORM*ABS(DT).LE.0.5) GO TO 15
   10 DT=DT/2.
      IND=IND+1
      GO TO 5
   15 T1=DT/2.
      L=0
      DO 25 I=1,NN,NDIM
      J=I+NM1
      DO 20 JJ=I,J
      X(JJ)=DT*A(JJ)
      Y(JJ)=X(JJ)
   20 B(JJ)=T1*Y(JJ)
      B(I+L)=B(I+L)+DT
   25 L=L+1
      DO 35 IT=3,15
      CALL MMUL(X,Y,N,N,N,Z)
      T1=T1/FLOAT(IT)
      DO 30 I=1,NN,NDIM
   30 CALL VADD(N,T1,B(I),Z(I))
   35 CALL TFR(Y,Z,N,N,1,1)
      IF (IND.EQ.0) RETURN
      CALL MEXP(N,A,DT,Y)
   40 CALL MMUL(Y,B,N,N,N,X)
      DO 45 I=1,NN,NDIM
   45 CALL VADD(N,1.,B(I),X(I))
      IND=IND-1
      IF (IND.EQ.0) RETURN
      CALL MMUL(Y,Y,N,N,N,X)
      CALL TFR(Y,X,N,N,1,1)
      GO TO 40
      END
```

SUBROUTINE TFR

```
      SUBROUTINE TFR(X,A,N,M,K,I)
C
C          I - 1  GIVES  X = A
C          I - 2  GIVES  X = A'
C          I - 3  GIVES  X = A AS A VECTOR
C          I - 4  GIVES  A = X    WHERE X WAS A VECTOR
C          I - 5  GIVES  DIAG A = X  WHERE X WAS A VECTOR
C
      DIMENSION A(1),X(1)
      COMMON/MAIN1/NDIM
      JS=(K-1)*NDIM*M
      JEND=M*NDIM
      GO TO (10,30,50,70,90),I
   10 DO 20 II=1,N
      DO 20 JJ=II,JEND,NDIM
   20 X(JJ)=A(JJ+JS)
      RETURN
   30 DO 40 II=1,N
      KK=(II-1)*NDIM
      DO 40 JJ=1,M
      LL=(JJ-1)*NDIM+II
   40 X(KK+JJ)=A(LL+JS)
      RETURN
   50 KK=0
      DO 60 II=1,JEND,NDIM
      LL=II+N-1
      DO 60 JJ=II,LL
      KK=KK+1
   60 X(KK)=A(JJ+JS)
      RETURN
   70 KK=M*N+1
      DO 80 II=1,M
      LL=(M-II)*NDIM+1
      DO 80 IJ=1,N
      KK=KK-1
      JJ=LL+N-IJ
   80 A(JJ+JS)=X(KK)
      RETURN
   90 SAVE=A(1)
      K=N
      DO 91 I=1,N
      L=N
      DO 92 J=1,N
      IK=(K-1)*NDIM+K
      X(IK)=0.
      IF(L.EQ.K)    X(IK)=A(L)
   92 L=L-1
   91 K=K-1
      X(1)=SAVE
      RETURN
      END
```

SUBROUTINE FACTOR

```
            SUBROUTINE FACTOR(N,X,A,MR)
C              MATRIX VIA CHOLESKY DECOMPOSITION WITH PIVOTING
            DIMENSION A(1), X(1), IP(20)
            COMMON/MSIM1/NDIM, NDIM1/INOU/KIN,KOUT
            TOL = 1.0E-6
            CALL TRANS1(N,X,A)
            NN = N*NDIM
            MR = 0
            I = 1
            DO 50  IO = 1,NN,NDIM1
            NEND = IO - I
            P=0.0
            IP(I)=I
            K = I
            DO 20  KK = IO,NN,NDIM1
            T = A(KK) - DOT2(NEND),A(K),A(K))
            IF(ABS(T).LE.(TOL*(ABS(A(KK))+TOL))) GO TO 20
            IF(ABS(T).LE.ABS(P)) GO TO 20
            IP(I) = K
            P = T
         20 K = K + 1
            IF(I.EQ.IP(I))  GO TO 25
            K = IP(I)
                    CALL SWAP(A(I), A(K), N, NDIM)
            K = IO + NDIM*(K-I)
            NP = N-I+1
                    CALL SWAP(A(IO),A(K),NP,1)
         25 NP = IO
            IF(P)  30, 35, 40
         30 MR = -1
            WRITE(KOUT,1000)
       1000 FORMAT(37H TRIED TO FACTOR AN INDEFINITE MATRIX)
            RETURN
         35 NP = NEND + N
         40 K = NEND + 1
            DO 42  KK = K,NP
         42 A(KK) = 0.0
            IF(P.EQ. 0.0)  GO TO 50
            T = SQRT(P)
            A(IO) = T
            MR = MR+1
            IF(I.EQ.N)  GO TO 55
            IO1 = IO + 1
            K = I + 1
            NP = NEND + N
            DO 43  KK = IO1,NP
            A(KK) = (A(KK) - DOT2(NEND, A(I), A(K)))/T
         43 K = K + 1
         50 I = I + 1
C                          UNSWAP ROWS
         55 DO 60  I = 1,N
            K = N - I +1
            KK = IP(K)
            IF(KK.EQ.K)  GO TO 60
                    CALL SWAP(A(K), A(KK), MR, NDIM)
         60 CONTINUE
                    CALL TRANS1(N,A,A)
            RETURN
            END
```

SUBROUTINE MEXP

```
      SUBROUTINE MEXP(N,A,T,EA)
      DIMENSION A(1),EA(1),C(15),P(15)
      COMMON/MAIN1/NDIM,NDIM1,Y(1)
      DATA P
     1 /1.,2.,2.,1.333333,6.666667E-1,2.666657E-1,8.8834665E-2,
     2 2.54E-2,6.349657E-3,1.40533E-3,2.812343E-4,5.575073E-5,
     3 9.232021E-6,0.,0./
      NN=NDIM*N
      NM1=N-1
      IF(T.NE.0.) GO TO 6
      CALL IDNT(N,EA,1.0)
      RETURN
    6 NTERM=13
      C1=XNORM(N,A)
      IND=0
      NTERM1=NTERM-1
      T1=T/2.0
   15 IF(ABS(T1*C1).LE.1.0) GO TO 20
      T1=T1/2.0
      IND=IND+1
      GO TO 15
   20 C1=1.0
      DO 22 I=1,NTERM
      C(I)=P(I)*C1
   22 C1=C1*T1
      C2=C(NTERM)
      C1=C(NTERM1)
      CALL DIAG(N,EA,A,C2,C1)
      DO 30 L=2,NTERM1
      C1=C(NTERM-L)
      CALL MMUL(A,EA,N,N,N,Y)
      CALL DIAG(N,EA,Y,1.0,C1)
   30 CONTINUE
      IF(IND.EQ.0) RETURN
      DO 100 L=1,IND
      CALL EQUATE(Y,EA,N,N)
  100 CALL MMUL(Y,Y,N,N,N,EA)
      RETURN
      END
```

SUBROUTINE EQUATE

```
      SUBROUTINE EQUATE(A,B,NR,NC)
      DIMENSION A(1),B(1)
      COMMON/MAIN1/NDIM
      NN=NC*NDIM
      NR1=NR-1
      DO 1 J=1,NN,NDIM
      II=J+NR1
      DO 1 I=J,II
      A(I)=B(I)
    1 CONTINUE
      RETURN
      END
```

SUBROUTINE MLINEQ

```
          SUBROUTINE MLINEQ(N,A,C,X,TOL1)
C         ANSWER RETURNED IN C AND X
          DIMENSION A(1),C(1),X(1)
          COMMON/MAIN1/ NDIM, NDIM1, F(1)
          COMMON/INOU/KIN,KOUT
          DT=.5
          DT1=0.
          NN=N*NDIM
          DO 5 II=1,NN,NDIM1
        5 DT1=DT1+ABS(A(II))
          DT1=DT1/N
          IF(DT1.GT.4.0) DT=DT*4.0/DT1
          II=1
          DO 20 I=1,N
          DO 15 JJ=I,NN,NDIM
       15 X(JJ)=DT*A(JJ)
          X(II)=X(II)-.5
       20 II=II+NDIM1
          CALL GMINV(N,N,X,F,MR,1)
          IF(MR.EQ.N) GO TO 21
          IT=0
          DO 18 I=1,NN,NDIM1
       18 C(I)=1.E25
          GO TO 95
       21 CALL MMUL(C,F,N,N,N,X)
C         INITIALIZATION OF X,F
          I=1
          DO 40 II=1,NN,NDIM
          J=II
          IF(I.EQ.1) GO TO 30
          DO 25 JJ=I,II,NDIM
          C(J)=C(JJ)
       25 J=J+1
       30 IO=J
          DO 35 JJ=II,NN,NDIM
          C(J)=DT*DOT(N,F(II),X(JJ))
       35 J=J+1
          F(IO)=F(IO)+1.0
       40 I=I+1
          GO TO 50
          ENTRY OLINEQ
          NN=N*NDIM
          CALL EQUATE(F,A,N,N)
       50 TOL=TOL1
          ITT=31
          IF(TOL1.GT.0.99)ITT=IFIX(TOL)
          IF(TOL1.GT.0.99)TOL=0.0
          ADV=TOL*1.E-7
          DO 99 IT=1,ITT
          NEZ=0
          SIZE=0.0
          CALL MMUL(C,F,N,N,N,X)
          I=1
          II=1
          J=1
          GO TO 75
       60 J=II
```

```
      DO 72 K=1,JM1
      IF(S(K).EQ.1.) GO TO 72
      TEMP=-DOT(NR,A(JJ),A(KK))
      CALL VADD(K,TEMP,U(JJ),U(KK))
   72 KK=KK+NDIM
   75 FAC=1./SQRT(FAC)
      DO 80 I=JJ,JEM
   80 A(I)=A(I)*FAC
      DO 85 I=JJ,JCM
   85 U(I)=U(I)*FAC
  100 JJ=JJ+NDIM
C     IF(MR.EQ.NR.OR.NR.EQ.NC) GO TO 120
C     IF(MT.EQ.0) GO TO 120
C 110 FORMAT(I3,10X,I2,4H OF RANK,I2)
  120 NEND=NC*NDIM
      JJ=1
      DO 135 J=1,NC
      DO 125 I=1,NR
      II=I-J
      S(I)=0.
      DO 125 KK=JJ,NEND,NDIM
  125 S(I)=S(I)+A(II+KK)*U(KK)
      II=J
      DO 130 I=1,NR
      U(II)=S(I)
  130 II=II+NDIM
  135 JJ=JJ+NDIM1
      RETURN
      END
```

SUBROUTINE DIAG

```
      SUBROUTINE DIAG(N,A,B,C1,C2)
      DIMENSION A(1),B(1)
      COMMON/MATRIX/NDIM,NDIM1
      NN=N*NDIM
      NM1=N-1
      II=1
      IF(C1.EQ.1.0) GO TO 10
      DO 5 J=1,NN,NDIM
      K=J+NM1
      DO 4 I=J,K
    4 A(I)=C1*B(I)
      A(II)=A(II)+C2
    5 II=II+NDIM1
      RETURN
   10 DO 7 J=1,NN,NDIM
      K=J+NM1
      DO 6 I=J,K
    6 A(I)=B(I)
      A(II)=A(II)+C2
    7 II=II+NDIM1
      RETURN
      END
```

SUBROUTINE INTEG

```
      SUBROUTINE INTEG(N,A,C,S,T)
C      S=INTEGRAL EA*C*EAT FROM 0 TO T   C IS EAT ON RETURN
      DIMENSION A(1),C(1),S(1),COEF(15)
      COMMON/MAIN1/ NDIM,NDIM1, X(1)
      NN=N*NDIM
      NT=14
      NTM1=NT-1
      IND=0
      ANORM=XNORM(N,A)
      DT=T
    5 IF (ANORM*ABS(DT).LE.1.0) GO TO 10
      DT=DT/2.
      IND=IND+1
      GO TO 5
   10 COEF(NT)=1.0
      DO 30 I=1,NTM1
      II=NT-I
   30 COEF(II)=DT*COEF(II+1)/FLOAT(I)
      CALL SCALE(S,C,N*N,COEF(1))
      IF(DT.EQ.0.) CALL IDNT(N,C,1.0)
      IF(DT.EQ.0.) RETURN
      DO 45 L=2,NTM1
      CALL MMUL(A,S,N,N,N,X)
      T1=COEF(L)
      DO 40 I=1,NN,NDIM1
      II=I
      DO 40 JJ=I,NN,NDIM
      S(JJ)=X(JJ)+X(II)+T1*C(JJ)
      S(II)=S(JJ)
   40 II=II+1
   45 CONTINUE
      CALL DIAG(N,C,A,COEF(1),COEF(2))
      DO 55 L=3,NT
      CALL MMUL(A,C,N,N,N,X)
      CALL DIAG(N,C,X,1.0,COEF(L))
   55 CONTINUE
   60 IF(IND.EQ.0) RETURN
      CALL MATR(N,N,C,S,X)
      DO 90 I=1,N
      DO 90 J=I,NN,NDIM
      S(J)=S(J)+X(J)
   90 X(J)=C(J)
      CALL MMUL(X,X,N,N,N,C)
      IND=IND-1
      GO TO 60
      END
```

SUBROUTINE IDNT

```
      SUBROUTINE IDNT(N,A,C1)
      DIMENSION A(1)
      COMMON/MAIN1/NDIM,NDIM1
      NN=N*NDIM
      II=1
      DO 1 I=1,N
      DO 2 J=I,NN,NDIM
    2 A(J)=0.0
      A(II)=C1
    1 II=II+NDIM1
      RETURN
      END
```

SUBROUTINE MMUL

```
      SUBROUTINE MMUL(X,Y,N1,N2,N3,Z)
      DIMENSION X(1),Y(1),Z(1)
      COMMON/MAIN1/NDIM
      N1M1=N1-1
      NN3=N3*NDIM
      DO 1 I=1,NN3,NDIM
      II=I+N1M1
      DO 1 J=I,II
    1 Z(J)=0.0
      ENTRY MMULS
      NN3=N3*NDIM
      KK=0
      DO 10 K=1,N2
      DO 8 I=1,N1
      C1=Y(I+KK)
      IF(C1.NE.0.0) CALL VADD1(NN3,C1,Z(I),X(K))
    8 CONTINUE
   10 KK=KK+NDIM
      RETURN
      END
```

SUBROUTINE TRANS1

```
      SUBROUTINE TRANS1(M,A,AT)
C     SETS AT= ATRANSPOSE A AND AT CAN BE EQUIVALENT, BOTH ARE,SQUARE
      DIMENSION A(1),AT(1)
      COMMON/MAIN1/NDIM,NDIM1
      NN=M*NDIM
      DO 1 I=1,NN,NDIM1
      IJ=I
      DO 1 JI=I,NN,NDIM
      TEMP=A(IJ)
      AT(IJ)=A(JI)
      AT(JI)=TEMP
      IJ=IJ+1
    1 CONTINUE
      RETURN
      END
```

SUBROUTINE VSCALE

```
      SUBROUTINE VSCALE(X,Y,N,C1)
      DIMENSION X(1),Y(1)
      L=0
      IF(C1.EQ.1.0) GO TO 5
      IF(C1.EQ.0.0) GO TO 8
    1 L=L+1
      X(L)=C1*Y(L)
      IF(L.LT.N) GO TO 1
      RETURN
    5 L=L+1
      X(L)=Y(L)
      IF(L.LT.N) GO TO 5
      RETURN
    8 L=L+1
      X(L)=0.0
      IF(L.LT.N) GO TO 8
      RETURN
      END
```

SUBROUTINE SCALE

```
      SUBROUTINE SCALE(A,B,NR,NC,C1)
      DIMENSION A(1),B(1)
      COMMON/MAIN1/NDIM
      NN=NC*NDIM
      NRM1=NR-1
      DO 2 I=1,NN,NDIM
      II=I+NRM1
      DO 2 J=I,II
    2 A(J)=C1*B(J)
      RETURN
      END
```

FUNCTION DOT

```
      FUNCTION DOT(NR,A,B)
      DOUBLE PRECISION DOT1,DBLE
      DIMENSION A(1),B(1)
      DOT1=0.D0
      IF(NR.LE.0) GO TO 2
      DO 1 I=1,NR
    1 DOT1=DOT1+DBLE(A(I)*B(I))
    2 DOT=DOT1
      RETURN
      END
```

FUNCTION DOT2

```
      FUNCTION DOT2(NN,A,B)
      DOUBLE PRECISION DDT2,DBLE
      DIMENSION A(1),B(1)
      COMMON/MAIN1/NDIM
      DDT2=0.D0
      IF(NN.LE.0) GO TO 2
      DO 1 I=1,NN,NDIM
    1 DDT2=DDT2+DBLE(A(I)*B(I))
    2 DOT2=DDT2
      RETURN
      END
```

FUNCTION DOT3

```
      FUNCTION DOT3(N,A,B)
      DOUBLE PRECISION DDT3,DBLE
      DIMENSION A(1),B(1)
      COMMON/MAIN1/NDIM
      DDT3=0.D0
      IF(N.LE.0) GO TO 2
      II=1
      DO 1 I=1,N
      DDT3=DDT3+DBLE(A(II)*B(I))
    1 II=II+NDIM
    2 DOT3=DDT3
      RETURN
      END
```

SUBROUTINE MAT2

```
                SUBROUTINE MAT2(N1,N2,X,Y,Z)
C               Z=(Y" Y,Y=N1*N2,Z=Z"
C               Z AND Y CAN BE EQUIVALENT
                DIMENSION X(1),Y(1),Z(1)
                COMMON/MAIN1/NDIM,NDIM1
                NN2=N2*NDIM
                II=1
                DO 10 I=1,N1
                IJ=II
                DO 5 J=I,N1
                Z(IJ)=DOT2(NN2,Y(I),Y(J))
              5 IJ=IJ+NDIM
                J=II
                IJ=J
              3 IJ=IJ-NDIM
                IF(IJ.LT.I) GO TO 10
                J=J-1
                Z(IJ)=Z(J)
                GO TO 3
             10 II=II+NDIM1
                RETURN
                END
```

SUBROUTINE MAT2A

```
                SUBROUTINE MAT2A(N1,N2,X,Y,Z)
C                 Z=YT*Y   X,Y ARE N1XN2,   Z=ZT
C                Z AND Y CAN BE EQUIVALENT
                DIMENSION X(1),Y(1),Z(1)
                COMMON/MAIN1/NDIM
                NN2=N2*NDIM
                I=0
                DO 10 II=1,NN2,NDIM
                J=II+I
                IJ=J-1
                DO 5 JJ=II,NN2,NDIM
                Z(J)=DOT(N1,X(II),Y(JJ))
              5 J=J+1
                I=I+1
                JJ=I
                DO 10 J=II,IJ
                Z(J)=Z(JJ)
             10 JJ=JJ+NDIM
                RETURN
                END
```

SUBROUTINE MAT3

```
      SUBROUTINE MAT3(N1,N2,X,Y,Z)
      DIMENSION X(1),Y(1),Z(1)
C      Z=XYXT    Y=YT IS N2XN2     X IS N1XN2
      CALL MMUL(X,Y,N1,N2,N2,Z)
      CALL MAT2(N1,N2,X,Z,Z)
      RETURN
      END
```

SUBROUTINE MAT3A

```
      SUBROUTINE MAT3A(N1,N2,X,Y,Z)
      DIMENSION X(1),Y(1),Z(1)
      CALL MMUL(Y,X,N2,N2,N1,Z)
      CALL MAT2A(N2,N1,X,Z,Z)
      RETURN
      END
```

SUBROUTINE MAT5A

```
      SUBROUTINE MAT5A(X,Y,N1,N2,N3,Z)
C        Z=XT*Y    X=N2*N1,  Y=N2*N3
      DIMENSION X(1),Y(1),Z(1)
      COMMON/MAIN1/NDIM
      N1M1=N1-1
      NN3=N3*NDIM
      DO 1 I=1,NN3,NDIM
      II=I+N1M1
      DO 1 J=I,II
    1 Z(J)=0.0
      ENTRY MAT5AS
      NN3=N3*NDIM
      DO 10 K=1,N2
      KK=K
      DO 8 I=1,N1
      C1=X(KK)
      IF(C1.NE.0.0) CALL VADD1(NN3,C1,Z(I),Y(K))
    8 KK=KK+NDIM
   10 CONTINUE
      RETURN
      END
```

SUBROUTINE VADD

```
      SUBROUTINE VADD(N,C1,A,B)
      DIMENSION A(1),B(1)
      DO 1 I=1,N
    1 A(I)=A(I)+C1*B(I)
      RETURN
      END
```

SUBROUTINE VADD1

```
      SUBROUTINE VADD1(NN,C1,A,B)
      DIMENSION A(1),B(1)
      COMMON/MAIN1/NDIM
      DO 1 I=1,NN,NDIM
    1 A(I)=A(I)+C1*B(I)
      RETURN
      END
```

SUBROUTINE MSUB

```
      SUBROUTINE MSUB(X,Y,NR,NC,Z)
      DIMENSION X(1),Y(1),Z(1)
      COMMON/MAIN1/NDIM
      NN=NC*NDIM
      DO 1 I=1,NN,NDIM
      JEND=I+NR-1
      DO 1 J=I,JEND
    1 Z(J)=X(J)-Y(J)
      RETURN
      END
```

SUBROUTINE MADD1

```
      SUBROUTINE MADD1(NR,NC,X,Y,Z,C1)
C     Z=X+C1*Y
      DIMENSION X(1),Y(1),Z(1)
      COMMON/MAIN1/NDIM
      NN=NC*NDIM
      IF(C1.EQ.1.)GO TO 1
      IF(C1.EQ.-1.)GO TO 2
      DO 5 I=1,NR
      DO 5 J=I,NN,NDIM
    5 Z(J)=X(J)+C1*Y(J)
      RETURN
    1 DO 10 I=1,NR
      DO 10 J=I,NN,NDIM
   10 Z(J) = X(J)+Y(J)
      RETURN
    2 DO 15 I=1,NR
      DO 15 J=I,NN,NDIM
   15 Z(J)=X(J)-Y(J)
      RETURN
      END
```

SUBROUTINE MADD

```
      SUBROUTINE MADD(X,Y,NR,NC,Z)
      DIMENSION X(1),Y(1),Z(1)
      COMMON/MAIN1/NDIM
      NN=NC*NDIM
      DO 1 I=1,NN,NDIM
      JEND=I+NR-1
      DO 1 J=I,JEND
    1 Z(J)=X(J)+Y(J)
      RETURN
      END
```

SUBROUTINE CDIV

```
      SUBROUTINE CDIV(AR,AI,BR,BI,CR,CI)
C         CR + I CI  =  (AR + I AI)/(BR + I BI)
      COMMON/INOU/KIN,KOUT
      GAMA=1.E-14
      IF(ABS(BR).GT.GAMA.OR.ABS(BI).GT.GAMA) GO TO 10
      WRITE (KOUT,100)
  100 FORMAT(//10X,35H ATTEMPTED COMPLEX DIVISION BY ZERO//)
      CALL  EXIT
   10 IF(ABS(BR).LE.ABS(BI)) GO TO 20
      T=BI/BR
      BC=1./(T*BI+BR)
      CR=(AR+T*AI)*BC
      CI=(AI-T*AR)*BC
      RETURN
   20 T=BR/BI
      BC=1./(T*BR+BI)
      CR=(T*AR+AI)*BC
      CI=(T*AI-AR)*BC
      RETURN
      END
```

SUBROUTINE SWAP

```
      SUBROUTINE SWAP(A,B,N,INC)
      DIMENSION A(1),B(1)
      NN=N*INC
      I=1
    1 IF(I.GT.NN) RETURN
      TEMP=A(I)
      A(I)=B(I)
      B(I)=TEMP
      I=I+INC
      GO TO 1
      END
```

SUBROUTINE REDCT

```
      SUBROUTINE REDCT(N,A,KK,KI,KL,M,D,NR,Q)
      DIMENSION A(1),D(1),Q(1),S(80)
      COMMON/MAIN1/NDIM,MDIM
      GAMA=1.E-14
      NN=N*NDIM
      T=SQRT(DOT(M,D,D))
      IF(T.LE.GAMA) GO TO 60
      T=SIGN(T,D(1))
      D(1)=D(1)+T
      DD=D(1)
      DO 20 I=1,M
      S(I)=D(I)/T
   20 D(I)=D(I)/DD
C        ROW AND COL MOD OF A
      KJ=KK
      IF(KI.GT.KK) KJ=KJ+NDIM
      DO 30 I=KJ,NN,NDIM
      P=-DOT(M,D,A(I))
   30 CALL VADD(M,P,A(I),S)
      DO 40 I=KI,KL
      P=DOT3(M,A(I),S)
      L=I
      DO 40 J=1,M
      A(L)=A(L)-D(J)*P
   40 L=L+NDIM
C        COLUMN MODIFICATION OF Q
      IF(NR.EQ.0) GO TO 60
      KJ=KI+N-1
      DO 50 I=KI,KJ
      P=DOT3(M,Q(I),S)
      L=I
      DO 50 J=1,M
      Q(L)=Q(L)-D(J)*P
   50 L=L+NDIM
   60 IF(KI.LE.KK) RETURN
      D(1)=-T
      DO 70 I=2,M
   70 D(I)=0.
      RETURN
      END
```

SUBROUTINE EIGEN

```
                  SUBROUTINE EIGEN(N,AA,RR,RI,Q,NR)
                  DIMENSION AA(1),RR(1),RI(1),Q(1),QR(3)
                  COMMON/MAIN1/NDIM,MDIM,A(1)/INOJ/KIN,KOUT
                  EQUIVALENCE (P,QR(1)),(B,QR(2)),(R,QR(3))
                  GAMAD=1.E-7
                  NN=N+(N-1)*NDIM
                  DO 2 I=1,N
                  DO 1 J=I,NN,NDIM
                1 A(J)=AA(J)
                2 RI(I)=0.0
                  IF(N.LE.2) GO TO 30
                  NE=(N-2)*NDIM
                  M=N-1
                  KI=MDIM
                  KN=N
                  DO 20 KK=2,NE,MDIM
                  KL=1+KK
                  KM=KN+NDIM
                  X=ABS(A(KK))
                  Y=X+GAMAD
                  DO 5 I=KL,KN
                5 X=X+ABS(A(I))
                  IF(X.GE.Y) GO TO 15
                  DO 5 I=KL,KN
                6 A(I)=0.0
                  GO TO 16
               15 CALL REDCT(N,A,KK,KI,KM,M,A(KK),NR,Q)
               16 KN=KM
                  KI=KI+NDIM
               20 M=M-1
C-----      A IS IN UPPER HESSENBERG
               30 MM=NN
                  M=N
                  TT=0.
                  NE=NN-NDIM
C           NEXT ITERATION
 40               NA=NE-MDIM
                  NS=MM-MDIM
                  GAMA=GAMAD
               45 ITS=0
                  X=A(MM)
                  IF(M.EQ.1) GO TO 210
C           LOOK FOR ZERO ON SUB-DIAGONAL
               50 J=MM
               60 TOL=GAMA*(ABS(A(J))+ABS(A(J-MDIM))                 +GAMA)
                  IF(ABS(A(J-NDIM)).LE.TOL) GO TO 70
                  J=J-MDIM
                  IF(J.GT.1) GO TO 60
 70               X=A(MM)
                  IF(J.EQ.MM) GO TO 210
                  Y=A(NS)
                  W=A(NE)*A(MM-1)
                  IF(J.EQ.NS) GO TO 220
                  IF(ITS.EQ.30) GO TO 300
                  IF(ITS.NE.20.AND.ITS.NE.10) GO TO 33
                  TT=TT+X
                  DO 83 I=1,MM,MDIM
```

SUBROUTINE EIGEN

```
   80  A(I)=A(I)-X
       Z=ABS(A(NE))+ABS(A(NA))
       X=.75*Z
       Y=X
       W=-.4375*Z*Z
   90  ITS=1+ITS
C         LOOK FOR TWO CONSECUTIVE SMALL SUB-DIAGONAL ELEMENTS
       K=NS-NDIM
  100  Z=A(K)
       R=X-Z
       S=Y-Z
       P=(R*S-W)/A(1+K)+A(K+NDIM)
       B=A(K+NDIM)-Z-R-S
       R=A(K+NDIM+2)
       IF(K.EQ.J) GO TO 110
       TOL=GAMA*ABS(P)*(ABS(A(K-NDIM))+ABS(Z)+ABS(A(K+NDIM)))
       IF(ABS(A(K-NDIM))*(ABS(B)+ABS(R)).LE.TOL) GO TO 110
       K=K-NDIM
       GO TO 100
C         BEGIN ITERATION WITH DIAGONAL ELEMENT K
  110  KI=(K+NDIM+1)/NDIM
       L=3
       KL=M-1+KI
       DO 200 KK=K,NS,NDIM
       IF(KK.EQ.NS) L=2
       KC=KK+L
       IF(KC.GT.KL) KC=KL
       IF(KK.NE.K) GO TO 150
       IF(J.NE.K) A(K-NDIM)=-A(K-NDIM)
       CALL REDCT(N,A,KK,KI,KC,L,QR,NR,0)
       GO TO 175
  150  KN=KK-NDIM
       CALL REDCT(N,A,KN,KI,KC,L,A(KN),NR,0)
  175  KL=KL+NDIM
  200  KI=KI+NDIM
       GO TO 50
C         FOUND ONE ROOT
  210  A(MM)=X+TT
       RR(M)=X+TT
       IF(M.EQ.1) GO TO 320
       M=M-1
       A(NE)=0.
       NE=NA
       MM=NS
       GO TO 40
C         FOUND TWO ROOTS
  220  P=.5*(Y-X)
       B=P*P+W
       Z=SQRT(ABS(B))
       X=X+TT
       A(MM)=X
       A(NS)=Y+TT
       IF(B.LT.0.) GO TO 290
C         REAL ROOTS
       Z=P+SIGN(Z,P)
  240  RR(M-1)=X+Z
       RR(M)=X-W/Z
```

SUBROUTINE EIGEN

```
            X=A(NE)
      C        REDUCE TO DIAGONAL
            Y=SQRT(X*X+Z*Z)
      C        REDUCE TO DIAGONAL
            Y=SQRT(X*X+Z*Z)
            P=X/Y
            B=Z/Y
      C        ROW AND COLUMN MOD OF A
            DO 250 I=NS,NN,NDIM
            Z=A(I)
            Y=A(I+1)
            A(I)=B*Z+P*Y
       250  A(I+1)=B*Y-P*Z
            KC=NE-M+1
            DO 260 I=KC,NE
            Z=A(I)
            Y=A(I+NDIM)
            A(I)=B*Z+P*Y
       260  A(I+NDIM)=B*Y-P*Z
      C        COLUMN MODIFICATION OF Q
            IF(NR.EQ.0) GO TO 275
            KK=KC+N-1
            DO 270 I=KC,KK
            Z=Q(I)
            Y=Q(I+NDIM)
            Q(I)=B*Z+P*Y
       270  Q(I+NDIM)=B*Y-P*Z
       275  M=M-1
            A(MM)=RR(M+1)
            A(NE)=0.
       280  IF(M.EQ.1) GO TO 320
            M=M-1
            A(NA)=0.
            NE=NA-MDIM
            MM=NS-MDIM
            GO TO 40
      C        COMPLEX ROOTS
       230  RR(M)=X+P
            RI(M)=-Z
            M=M-1
            RR(M)=X+P
            RI(M)=Z
            GO TO 280

       300  WRITE(KOUT,310)
       310  FORMAT(41H  ONE DIGIT ACCURACY DROP IN QR ALGORITHM)
            GAMA=GAMA*10.
            GO TO 45
       320  CONTINUE
            RETURN
            END
```

SUBROUTINE MRIC

```
      SUBROUTINE MRIC(N,A,S,Q,X,Z,IER)
      DIMENSION A(1),S(1),Q(1),X(1),Z(1),FR(80),TIMES(3)
      REAL Z6(12),Z7(12),Z8(144),Z9(12)
      INTEGER EIG
      COMMON/MAIN1/ NDIM,NDIM1,F(1)/MAIN2/G(1)/INOU/KIN,KOUT
      COMMON/BOMB/EIG
      DATA TIMES/2.0,0.5,4.0/
      NN=N*NDIM
      TOL=1.E-03
    1 T1=-.5*ALOG(XNORM(N,Q)+.0001)
      IF(T1.LT.1.) T1=1.
      T1=T1*4.*FLOAT(N)/(1.+XNORM(N,A))
      KEY=0
    5 KEY=KEY+1
   10 CALL EQUATE(X,S,N,N)
      CALL INTEG(N,A,X,Z,T1)
      IF (EIG.NE.1) THEN
         GO TO 6
      ENDIF
      DO 11 I=1,NN
         Z8(I)=0.
11    CONTINUE
      DO 22 I=1,NN,NDIM1
         Z8(I)=1.
22    CONTINUE
      CALL EIGEN(N,Z,Z7,Z6,Z8,1)
      DO 23 I=1,N
      Z9(I)=1/Z7(I)
23    CONTINUE
      PRINT '(/T2,9X,2A//14X,A,7X,A,14X,A/)',
     !       'EIGENVALUES OF CONTROLLABILITY/OBSERVABILITY',
     !       'GRAMIAN','REAL','IMAGINARY','RECIPROCAL'
      DO 33 I=1,N
      PRINT '(10X,2(E12.5,2X),10X,E12.5)',Z7(I),Z6(I),Z9(I)
33    CONTINUE
      CALL EGVEC(N,F,Z7,Z6,Z8)
      PRINT '(/T2,2A/)','EIGENVECTORS OF THE CONTROLLABILITY/',
     !                      'OBSERVABILITY GRAMMIAN'
      DO 44 I=1,N
         PRINT'(10(3X,E10.4))',(Z8(I+J),J=0,NN-NDIM,NDIM)
44    CONTINUE
6     CALL FACTOR(N,Z,F,MR)
      IF(MR.NE.N) WRITE(KOUT,15) N,N,MR
   15 FORMAT(12H GRAMMIAN IS,I2,1HX,I2,9H OF RANK ,I2)
      IF(MR.LT.0) GO TO 60
      CALL GMINV(N,N,F,Z,MR,0)
      CALL MAT5A(X,Z,N,N,N,F)
      CALL MAT2(N,N,F,F,X)
      GO TO 18
C     A+SX IS STABLE
      ENTRY MRIC1
      T1=-99999.99
      TOL=1.E-03
      KEY=5
   18 NN=N*NDIM
      NM1=N-1
      DO 19 I=1,N
```

SUBROUTINE MRIC

```
   19 TR(I)=-1.
      TOL1=.1*TOL
      MAXIT=20*N
      DO 40 IT=1,MAXIT
      CALL MMUL(S,X,N,N,N,F)
      CALL MAT2A(N,N,X,F,Z)
      DO 20 I=1,NN,NDIM
      II=I+NM1
      DO 20 J=I,II
      G(J)=A(J)-F(J)
   20 Z(J)=Z(J)+Q(J)
      CALL MLINEQ(N,G,Z,X,TOL1)
      CONV=0.0
      C1=0.0
      II=1
      DO 25 I=1,N
      CONV=CONV+ABS(X(II)-TR(I))
      TR(I)=X(II)
      II=II+NDIM1
   25 C1=C1+ABS(TR(I))
      IF(C1.GT.1.E20) GO TO 50
      IF(CONV.GT.C1*TOL) GO TO 40
      CALL GMINV(N,N,Z,F,MR,0)
      CALL MMUL(S,X,N,N,N,Z)
      WRITE(KOUT,27)IT
   27 FORMAT(17H0RICCATI SOLN IN ,I2,11H ITERATIONS)
      CALL MADD1(N,N,A,Z,Z,-1.0)
      IF(MR .EQ. N)RETURN
      WRITE(KOUT,35)MR
   35 FORMAT(26H0RICCATI SOLN IS PSD--RANK,I3)
      RETURN
   40 CONTINUE
      TOL=10.*TOL
      WRITE(KOUT,45)MAXIT,TOL
   45 FORMAT(20H0RICCATI NON-CONV IN,I2,18H ITERS. UP TOL TO ,1PE10.4)
      GO TO 18
   50 WRITE(KOUT,55)IT,T1
   55 FORMAT(29H0RICCATI BLOW UP AT ITERATION,I2,12H INITIAL T=,F10.5)
      IF(KEY.EQ.5) GO TO 1
50    IF (KEY.EQ.4) IER=1
      IF (KEY.EQ.4) RETURN
      T1=T1*TIMES(KEY)
      WRITE(KOUT,65)T1
   65 FORMAT(14H0RESET WITH T=,F10.5)
      GO TO 5
      END
```

Sample Program Output

```
WELCUME TO OPSACT. INDICATE INTERCOM MODE BY ENTERING A "1"

IN ORDER THAT THE MOTOR NOISE BE PROPERLY AUGMENTED INTO THE "W" MATRIX, THE "A" MATRIX MUST BE ARRANGED AS FOLLOWS:

   ***      .       .          ***
   *        .       .            *
   * BASIC  .       .            *
   * SYSTEM .       .            *
   *        .       .            *
   *.........       .            *
   *                .            *
   * AUGMENTING     .            *
   * CONTROL INPUTS .            *
   *                .            *
   *.................            *
   *                             *
   * AUGMENTING                  *
   * CONTROL SYSTEM              *
   *                             *
   ***                         ***


RUN/CONFIGURATION NUMBER IS: 15


THE   "A"   MATRIX IS:

  -.4100E-01   .1100E+00  -.2500E+02  -.3220E+02  0.          0.           .4000E+00   .3200E-02  0.
  -.2500E+00  -.7500E+00   .2050E+03   .1405E+01  0.          0.          0.           .1100E+01  0.
  0.          -.2321E-02  -.7600E+00  0.          0.          0.          0.           .3369E+00  0.
  0.          0.           .1000E+01  0.          0.          0.          0.          0.          0.
  -.4363E-01  -.1000E+01  0.           .2039E+03  0.          0.          0.          0.          0.
  0.          0.          0.          0.          0.          0.          0.          0.          0.
  0.          0.          0.          0.          0.          0.          0.          0.          0.
  0.          0.          0.          0.          0.          0.          0.          0.           .1000E+01
  0.          0.          0.          0.          0.           .2560E+03  0.          -.2560E+03  -.2240E+02
THE   "B"   MATRIX IS:

  0.          0.
  0.          0.
  0.          0.
  0.          0.
  0.          0.
   .1000E+01  0.
  0.           .1000E+01
  0.          0.
  0.          0.
```

THE "C" MATRIX IS:

0.	.4878E-02	0.	0.	0.	0.	0.	0.	0.
-.1220E-02	-.3659E-02	.1000E+01	.6854E-02	0.	0.	0.	.5366E-02	0.
0.	0.	0.	.1000E+01	0.	0.	0.	0.	0.
0.	0.	.1000E+01	0.	0.	0.	0.	0.	0.
-.4363E-01	-.1000E+01	0.	.2039E+03	0.	0.	0.	0.	0.
0.	0.	0.	0.	.2309E-04	0.	0.	0.	0.
.1000E+01	0.	0.	0.	0.	0.	0.	0.	0.

THE "Q" MATRIX IS:

.8765E-01	0.	0.	0.	0.	0.	0.	0.	0.
0.	.4000E-01	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	.1313E+03	0.	0.	0.	0.	0.
0.	0.	0.	0.	.4000E-01	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	.4000E+03	0.	0.
0.	0.	0.	0.	0.	0.	0.	.1313E+03	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.

THE "R" MATRIX IS:

.1100E+01	0.
0.	.4000E+03

THE "V" MATRIX IS:

.1013E-03	0.	0.	0.	0.	0.	0.
0.	.3186E-03	0.	0.	0.	0.	0.
0.	0.	.2362E-04	0.	0.	0.	0.
0.	0.	0.	.4972E-03	0.	0.	0.
0.	0.	0.	0.	.1498E+01	0.	0.
0.	0.	0.	0.	0.	.2133E-09	0.
0.	0.	0.	0.	0.	0.	.2402E+00

THE "W" MATRIX IS:

.4037E+01	-.2753E+02	-.8520E-01	0.	0.	0.	0.	0.	0.
-.2753E+02	.1877E+03	.5809E+00	0.	0.	0.	0.	0.	0.
-.8520E-01	.5809E+00	.1798E-02	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	.7162E-07	0.	0.	0.
0.	0.	0.	0.	0.	0.	.9837E-04	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.

THE INITIAL TIME IS: 0.00

THE FINAL TIME IS: 200.00

THE TIME INCREMENT IS: .025

THE XI MATRIX IS:

0.
0.
0.
0.
-.2000E+02
0.
0.
0.
0.

THE UI MATRIX IS:

0.
0.

THE THRSHLD MATRIX IS:

.2031E-02
.7311E-02
.1015E-02
.3655E-02
.2778E+00
.2670E-04
.8843E+00

RICCATI SOLN IN 14 ITERATIONS

THE K MATRIX IS:

.1493E+01	-.1281E+00	.3608E+01	.6825E+02	.2435E+00	.8864E-01	.5784E+00	.6784E-01	.3243E-02
-.1281E+00	.1741E+00	-.1183E+02	-.5804E+02	-.1369E+00	-.2955E+00	-.4392E-01	-.3052E+00	-.1120E-01
.3608E+01	-.1183E+02	.2015E+04	.5363E+04	.8895E+01	.5381E+02	.9689E+00	.5404E+02	.1993E+01
.6825E+02	-.5804E+02	.5363E+04	.2259E+05	.5117E+02	.1345E+03	.2483E+02	.1399E+03	.5128E+01
.2435E+00	-.1369E+00	.8895E+01	.5117E+02	.1652E+00	.2097E+00	.9308E-01	.2259E+00	.8237E-02
.8864E-01	-.2955E+00	.5381E+02	.1345E+03	.2097E+00	.1106E+02	.2250E-01	.3870E+01	.2172E+00
.5784E+00	-.4392E-01	.9689E+00	.2483E+02	.9308E-01	.2250E-01	.4002E+03	.2264E-01	.8330E-03
.8784E-01	-.3052E+00	.5404E+02	.1399E+03	.2259E+00	.3870E+01	.2264E-01	.9998E+01	.2997E+00
.3243E-02	-.1120E-01	.1993E+01	.5128E+01	.8237E-02	.2172E+00	.8330E-03	.2997E+00	.1242E-01

A (ONE)

-.4100E-01	.1100E+00	-.2500E+02	-.3220E+02	0.	0.	.4000E+00	.3200E-02	0.
-.2500E+00	-.7500E+00	.2050E+03	.1405E+01	0.	0.	0.	.1100E+01	0.
0.	-.2321E-02	-.7600E+00	0.	0.	0.	0.	.3369E+00	0.
0.	0.	.1000E+01	0.	0.	0.	0.	0.	0.
-.4363E-01	-.1000E+01	0.	.2039E+03	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	-.1006E+02	-.2045E-01	0.	0.
0.	0.	0.	0.	0.	-.5624E-04	-.1001E+01	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	.1000E+01
0.	0.	0.	0.	0.	.2560E+03	0.	-.2560E+03	-.2240E+02

B (ONE)

0.	0.
0.	0.
0.	0.
0.	0.
0.	0.
.1006E+02	.2045E-01
.5624E-04	.1001E+01
0.	0.
0.	0.

THE L (ZERO) MATRIX

.8058E-01	-.2686E+00	.4892E+02	.1223E+03	.1906E+00	0.	0.	.3518E+01	.1975E+00
.1446E-02	-.1098E-03	.2422E-02	.6207E-01	.2327E-03	0.	0.	.5660E-04	.2083E-05

OPEN LOOP EIGENVALUES

REAL	IMAGINARY
-.74931E+00	.69080E+00
-.74931E+00	-.69080E+00
-.26185E-01	.13204E+00
-.26185E-01	-.13204E+00
.33840E-08	0.
-.11200E+02	.11426E+02
-.11200E+02	-.11426E+02
-.10055E+02	0.
-.10006E+01	0.

CLOSED LOOP EIGENVALUES

REAL	IMAGINARY
-.82733E+01	.24414E+02
-.82733E+01	-.24414E+02
-.19214E+02	0.
-.49142E+01	0.
-.32926E-01	0.
-.70800E+00	.56792E+00
-.70800E+00	-.56792E+00
-.98280E+00	0.
-.10000E+01	0.

RICCATI SOLN IN 12 ITERATIONS

RICCATI SOLN IN 9 ITERATIONS

RICCATI SOLN IN 9 ITERATIONS
STATE COVARIANCE MATRIX

.1242E+03	-.7207E+02	.1593E-01	-.3563E+00	.9282E+02	-.2466E+00	-.1824E+00	-.2438E+00	-.2845E-01
-.7207E+02	.2499E+03	.3591E+00	.6047E+00	-.2053E+03	.9790E+00	.8841E-01	.9618E+00	.1932E+00
.1593E-01	.3591E+00	.1068E-02	-.1557E-09	-.2686E-01	.2252E-02	-.7577E-04	.2216E-02	.3814E-03
-.3563E+00	.6047E+00	-.1557E-09	.3021E-02	-.9867E+00	.8013E-03	.5397E-03	.9967E-03	-.2216E-02
.9282E+02	-.2053E+03	-.2686E-01	-.9867E+00	.3439E+03	-.3389E+00	-.1423E+00	-.4068E+00	.7480E+00
-.2466E+00	.9790E+00	.2252E-02	.8013E-03	-.3389E+00	.1081E-01	.2582E-03	.1003E-01	.1045E-01
-.1824E+00	.8841E-01	-.7577E-04	.5397E-03	-.1423E+00	.2582E-03	.2744E-03	.2773E-03	-.2070E-03
-.2438E+00	.9618E+00	.2216E-02	.9967E-03	-.4068E+00	.1003E-01	.2773E-03	.1050E-01	.3913E-10
-.2845E-01	.1932E+00	.3814E-03	-.2216E-02	.7480E+00	.1045E-01	-.2070E-03	.3913E-10	.1194E+00

THE P MATRIX IS:

.6140E+00	-.2834E+01	-.6990E-02	-.1970E-02	.5359E-02	.2473E-04	.7309E-06	.6357E-04	-.4962E-03
-.2834E+01	.1876E+02	.4904E-01	.7255E-02	-.1819E+01	-.1756E-03	.6502E-07	-.4532E-03	.3728E-02
-.6990E-02	.4904E-01	.1439E-03	.1013E-04	-.1740E-02	.5802E-05	-.5228E-09	.1285E-04	-.7277E-04
-.1970E-02	.7255E-02	.1013E-04	.2301E-04	.2602E-03	.6382E-06	.2746E-09	.1562E-05	-.1412E-04
.5359E-02	-.1819E+01	-.1740E-02	.2602E-03	.4788E+01	-.2783E-04	.2649E-06	-.6769E-04	.5187E-03
.2473E-04	-.1756E-03	.5802E-05	.6382E-06	-.2783E-04	.5083E-03	-.4957E-08	.2233E-03	.2249E-02
.7309E-06	.6502E-07	-.5228E-09	.2746E-09	.2649E-06	-.4957E-08	.1296E-05	-.6004E-08	.5897E-08
.6357E-04	-.4532E-03	.1285E-04	.1562E-05	-.6769E-04	.2233E-03	-.6004E-08	.3232E-03	.4524E-05
-.4962E-03	.3728E-02	-.7277E-04	-.1412E-04	.5187E-03	.2249E-02	.5897E-08	.4524E-05	.2565E-01

THE KALMAN FILTER GAIN MATRIX (S) IS:

-.7089E+02	.4704E+02	-.2036E+02	-.1756E+03	.5756E+00	.2024E+02	.1382E+0
.4693E+03	-.2893E+03	.7498E+02	.1232E+04	-.4106E+01	-.6868E+04	-.6379E+0
.1227E+01	-.4833E+00	.1047E+00	.3614E+01	-.1117E-01	-.6570E+01	-.1574E-0
.1815E+00	-.2489E+00	.2378E+00	.2546E+00	-.5927E-03	.9827E+00	-.4436E-0
-.4550E+02	.8829E+02	.2689E+01	-.4370E+02	.4478E+00	.1808E+05	.1206E-0
-.4394E-02	.1370E+00	.6595E-02	.1457E+00	.7291E-04	-.1051E+00	.5568E-0
.1626E-05	-.3026E-04	.2838E-05	-.1313E-04	-.9790E-08	.1000E-02	.1645E-0
-.1134E-01	.2910E+00	.1615E-01	.3228E+00	.1840E-03	-.2556E+00	.1431E-0
.9325E-01	-.1545E+01	-.1459E+00	-.1828E+01	-.1576E-02	.1959E+01	-.1117E-0

CLOSED-LOOP ESTIMATOR EIGENVALUES

REAL	IMAGINARY
-.11197E+02	.11426E+02
-.11197E+02	-.11426E+02
-.10559E+02	0.
-.10077E+02	0.
-.13956E+00	0.
-.30368E+00	.19526E+00
-.30368E+00	-.19526E+00
-.14004E+01	0.
-.10006E+01	0.

THE VALUE OF THE PERFORMANCE INDEX IS: .294013E+04

Appendix D

CALSPAN Data

Body Axis Dimensional Stability Derivatives

```
******************** LONGITUDINAL INPUTS *********
 XU= -4.1000E-02   XW=  1.1000E-01   XQ=  0.0          WO=  2.5000E+01   XT=  0.0          XD1=  3.2000E-03
 ZU= -2.5000E-01   ZW= -7.5000E-01   ZQ=  0.0          UO=  2.0500E+02   ZT=  0.0          ZD1=  1.1000E+00
 MU=  0.0          MW= -2.3213E-03   MQ= -7.6000E-01   MT=  0.0          MD1=  3.3685E-01  THET=  4.5000E+00
 XD2=  0.0         XD3=  0.0         ZD2=  0.0         ZD3=  0.0         MD2=  0.0         MD3=  0.0

OUTPUT FORMAT 1
   THE  CHARACTERISTIC  EQUATION (IN DESCENDING POWERS OF  S)

       1.0000E+00    1.5510E+03    1.1353E+00    7.2429E-02    1.8372E-02

   REAL          IMAGINARY       ZETA          OMEGA          TAU

 -7.5358E-01    6.9559E-01    7.3481E-01    1.0255E+00
 -2.1917E-02    1.3334E-01    1.6582E-01    1.3217E-01
```

Mode Ratios (Eigenvectors)

```
 V   /U          V   -DEG        THET/U         THET-DEG

  2.4914E+01    1.4635E+02    8.1066E-02    -8.0415E+01
  1.8179E-01    1.7685E+02    4.2599E-03     6.7285E+01
```

```
*********************THE TRANSFER FUNCTIONS*****

   REAL          IMAGINARY       ZETA          OMEGA          TAU

 U   /D1   =    3.20000E-03   -8.29534E+00   -9.36447E+00   -8.11437E+00
  2.5934E 03                                               -3.8559E-04
 -5.6439E-01   -8.1193E-01    5.7077E-01    9.8882E-01

 W   /D1   =    1.10000E+00    6.99339E+01    4.11992E+00    2.66605E+00
 -6.3518E+01                                                1.5744E-02
 -2.9183E-02   -1.9315E-01    1.4939E-01    1.9534E-01

 THETA/D1  =    3.36847E-01    2.63892E-01    1.95185E-02
 -8.3672E-02                                                1.2093E+01
 -7.0077E-01                                                1.4271E+00
```

u/δ_{ES}

w/δ_{ES}

θ/δ_{ES}

δ_{ES}

Numerator Polynomials and Factors

LONGITUDINAL CHARACTERISTICS, CONFIGURATION 1-1, V_{ind}=120 KTS

```
********************** LONGITUDINAL INPUTS *********
                                                          WO=  2.5000E+01   XT=  0.0          XD1=  3.2000E-03
 XU= -4..200E-02    XW=  1.1000E-01   XQ=  0.0
                                                          UO=  2.0500E+02   ZT=  0.0          ZD1=  1.1000E+00
 ZU= -2.6000E-01    ZW= -8.0000E-01   ZQ=  0.0
                                                          MT=  0.0          MD1=  3.3685E-01  THET=  4.5000E+00
 MU=  0.0           MW= -1.8745E-02   MQ= -1.8300E+00
                                                          ZD3=  0.0         MD2=  0.0         MD3=  0.0
 XD2=  0.0          XD3=  0.0         ZD2=  0.0

OUTPUT FORMAT 1 THE  CHARACTERISTIC  EQUATION  IN DESCENDING POWERS OF  S

          1.0000E+00   2.6710E+00   5.4431E+00   3.4444E-01   1.5437E-01
                                                                  W    /U        W    -DEG      THET/U       THET-DEG
   REAL        IMAGINARY      ZETA        OMEGA        TAU
                                                                  1.1825E+01   -1.6038E+02   4.9257E-02   2.5492E-01
 -1.3101E+00   1.8830E+00   5.7011E-01   2.2980E+00              8.8904E-02    1.7766E+02   5.3775E-03   7.3775E+01
 -2.5359E-02   1.6908E-01   1.4832E-01   1.7097E-01
*********************THE TRANSFER FUNCTIONS*****
   REAL        IMAGINARY      ZETA         OMEGA          TAU
 U   /D1  =    3.20000E-03  -8.29175E+00  -9.19077E+00  -8.07526E+00
                                                        -3.8576E-04
  2.5923E+03
 -5.5416E-01  -8.1632E-01    5.6166E-01    9.8665E-01
                                           4.25147E+00   2.77408E+00
               1.10000E+00   7.11108E+01                 1.5483E-02
 W   /D1  =
 -6.4587E+01
 -2.9618E-02  -1.9537E-01    1.4939E-01    1.9780E-01
               3.36847E-02   2.62669E-01   1.98526E-02   1.1792E+01
 THET/D1  =                                              1.4309E+00
 -8.4803E-02
 -6.9490E-01
```

LONGITUDINAL CHARACTERISTICS, CONFIGURATION 2-1, $V_{ind.}$=120 KTS

```
******************** LONGITUDINAL INPUTS *********
                                                            WO=  2.5000E+01   XT=  0.0           XD1=  3.2000E-03
XU= -4.1000E-02   XW=  1.1000E-01   XQ=  0.0                UO=  2.0500E+02   ZT=  0.0           ZD1=  1.1000E+00
ZU= -2.6000E-01   ZW= -8.1000E-01   ZQ=  0.0                MT=  0.0          MD1=  3.3695E-01   THET=  4.5000E+00
MU=  0.0          MW= -2.2881E-02   MQ=  2.1000E-01         ZD3=  0.0         MD2=  0.0          MD3=  0.0
XD2=  0.0         XD3=  0.0         ZD2=  0.0

UTPUT FORMAT : THE  CHARACTERISTIC  EQUATION  IN DESCENDING POWERS OF  S

           1.0000E+00   6.4100E-01   4.5737E+00   2.7031E-01   1.8844E-01
                                                                                                           THET-DEG
                                                               W   /U       W   -DEG      THET/U
   REAL        IMAGINARY     ZETA        OMEGA        TAU
                                                               9.4851E+00   -1.4631E+02   4.7359E-02    1.??8?E+01
-2.9319E-01   2.1?59E+00    1.3822E-01   2.1213E+00            1.7963E-02   -4.5474E+01   6.4341E-03    7.?376E+01
-2.7307E-02   2.0281E-01    1.3344E-01   2.0464E-01
*************************THE TRANSFER FUNCTIONS*****
                                                           TAU
   REAL        IMAGINARY      ZETA          OMEGA
                                                                         
U   /D1   =    3.20000E-03   -8.29825E+00   -9.41059E+00   -8.03740E+00
 2.5943E+03                                                -3.8545E-04
-5.6696E-01   -8.0417E-01    5.7621E-01     9.8334E-01
                                                           2.77408E+00
W   /D1   =    1.?0000E+00    6.88668E+01    4.16116E+00   1.5988E-02
-6.2546E+01
-2.9318E-02   -1.9856E-01    1.4900E-01     2.0080E-01

THET/D1   =    3.36847E-01    2.61487E-01    1.98076E-02   1.1755E+01
-8.5073E-02                                                1.4467E+00
-6.9121E-01
```

LONGITUDINAL CHARACTERISTICS, CONFIGURATION 3-0, V_{ind}=120 KYS

```
********************* LONGITUDINAL INPUTS *********

 XU= -4.1000E-02    XW=  1.1000E-01   XQ=  0.0          WO=  2.5000E+01   XT=  0.0          XD1=  3.2000E-03
 ZU= -2.6000E-01    ZW= -7.5000E-01   ZQ=  0.0          UO=  2.0500E+02   ZT=  0.0          ZD1=  1.1000E+00
 MU=  0.0           MW= -5.3267E-03   MQ= -4.4510E+00   MT=  0.0          MD1=  3.3685E-01  THET=  4.5000E+00
XD2=  0.0          XD3=  0.0          ZD2=  0.0         ZD3=  0.0         MD2=  0.0         MD3=  0.0

OUTPUT FORMAT :
                 THE  CHARACTERISTIC  EQUATION  IN DESCENDING POWERS OF  S

                 1.0000E+00   5.2420E+00   4.6721E+00   3.3012E-01   4.3858E-02

    REAL         IMAGINARY     ZETA         OMEGA         TAU         W  /U         W  -DEG       THET/U       THET-DEG

 -1.0486E+00                                              9.5367E-01   9.9511E+00   -1.8000E+02   1.4857E-02   0.0
 -4.1293E+00                                              2.4217E-01   1.0341E+01   -1.8000E+02   4.1467E-02   0.0
 -3.2063E-02     9.5411E-02    3.1854E-01   1.0065E-01                 2.6467E-01    1.7307E+02   3.1689E-03   5.3259E+01

**********************THE TRANSFER FUNCTIONS*****

    REAL         IMAGINARY     ZETA          OMEGA         TAU

 U   /D1   =     3.20000E-03   -8.28353E+00  -8.82438E+00  -8.00836E+00
  2.5097E+03                                               -3.8615E-04
 -5.3261E-01    -8.2026E-01    5.4190E-01    9.8305E-01

 W   /D1   =     1.10000E+00   7.39929E+01 ,  4.36750E+00   2.77408E+00
 -6.7209E+01                                               1.4879E-02
 -2.9259E-02    -1.9149E-01    1.5105E-01    1.9371E-01

 THET/D1   =     3.36847E-01   2.60586E-01   1.97560E-02
 -8.5196E-02                                               1.1738E+01
 -6.8841E-01                                               1.4626E+00
```

LONGITUDINAL CHARACTERISTICS, CONFIGURATION 4-0, $V_{ind.}$=120 KTS

```
******************** LONGITUDINAL INPUTS *********

 XU= -4.1000E-02   XW=  1.1000E-01   XQ=  0.0           WO=  2.5000E+01   XT=  0.0          XD1=  3.2000E-03

 ZU= -2.6000E-01   ZW= -9.2000E-01   ZQ=  0.0           UO=  2.0500E+02   ZT=  0.0          ZD1=  1.1000E+00

 MU=  0.0          MW= -5.9341E-02   MQ= -3.2500E+00    MT=  0.0         MD1=  3.3685E-01  THET=  4.5000E+00

XD2=  0.0         XD3=  0.0         ZD2=  0.0          ZD3=  0.0         MD2=  0.0          MD3=  0.0

OUTPUT FORMAT :
          THE  CHARACTERISTIC  EQUATION  IN DESCENDING POWERS OF  S

          1.0000E+00   4.2110E+00   1.5354E+01   9.5023E-01   4.8870E-01

   REAL         IMAGINARY      ZETA         OMEGA          TAU        W   /U         W   -DEG      THET/U        THET-DEG

 -2.0785E+00    3.2829E+00   5.3493E-01   3.8855E+00                9.9658E+00   -1.6558E+02   4.3665E-02   1.7226E+00
 -2.7014E-02    1.7798E-01   1.5015E-01   1.7992E-01                5.5156E-02    1.7729E+02   5.6358E-03   7.5498E+01
**********************THE TRANSFER FUNCTIONS*****

   REAL         IMAGINARY      ZETA         OMEGA          TAU

 U   /D1   =     3.20000E-03  -8.28603E+00  -8.88159E+00  -7.93979E+00
  2.5907E+03                                             -3.8600E-04
 -5.3585E-01   -8.1890E-01   5.4755E-01   9.7863E-01

 W   /D1   =     1.10000E+00   7.26728E+01   4.31433E+00   2.77408E+00
 -6.6007E+01                                              1.5150E-02
 -2.9420E-02   -1.9324E-01   1.5052E-01   1.9546E-01

 THET/D1   =     3.36847E-01   2.58435E-01   1.97128E-02
 -8.5894E-02                                              1.1642E+01
 -6.8132E-01                                              1.4677E+00
```

LONGITUDINAL CHARACTERISTICS, CONFIGURATION 5-1, V_{ind}=120 KTS

2-1	CONFIGURATION ω_{SP} = 2.3 ζ_{SP} = 0.57 --/ -- /--	FLIGHT/PILOT 1883/A
PILOT RATING: OVERALL --	APPROACH 3 PIO 1	SP 1
WIND/X-WIND: 0/04	TURBULENCE: None	$M_{\delta_{ES}}$ = 0.26

FEEL:

- Forces: Little heavy, not uncomfortable, okay.
- Displacement: No comments.
- Sensitivity: No comments.

PITCH ATTITUDE RESPONSE:

- Initial: Initial response okay, some tendency to overshoot, not a problem.
- Predictability: Tendency to overshoot one time when changing pitch attitude aggressively. For most of the task it was not a problem.
- Special Inputs: No comments.
- PIO Tendency: No comments.

AIRSPEED CONTROL: Adequate.

PERFORMANCE:

- Approach Tasks:
  ILS: Satisfactory.

  Visual (Sidestep): No comments.
- Landing tasks: Not done.
- Differences: No comments.

WIND AND TURBULENCE: Not a factor.

SUMMARY COMMENTS: Annoying overshoot tendency in pitch but this problem did not affect task performance.

5-3	CONFIGURATION $\omega_{SP} = 3.9$ $\zeta_{SP} = 0.54$ -- / 0.25 / --	FLIGHT/PILOT 1890/A
PILOT RATING: OVERALL -- APPROACH 4 PIO 1 SP 4½		
WIND/X-WIND: 04/03 TURBULENCE: Light $M_{\delta_{ES}} = 1.0$		

FEEL:
- Forces: Comfortable forces.
- Displacement: No comments.
- Sensitivity: Sensitivity okay.

PITCH ATTITUDE RESPONSE:
- Initial: Responded in desired fashion.
- Predictability: Predictable enough.
- Special Inputs: Very responsive to turbulence, annoying, increased workload.
- PIO Tendency: No special inputs or PIO tendency.

AIRSPEED CONTROL: Pretty good.

PERFORMANCE:
- Approach Tasks:
  ILS: Went well.

  Visual (Sidestep): Appears okay.
- Landing tasks: Not done.
- Differences: Not applicable.

WIND AND TURBULENCE: Turbulence noticeable and is a factor.

SUMMARY COMMENTS: Annoying aircraft to fly, hard to evaluate. Tendency in pitch to wander off a problem, tended to come back.

5-3	CONFIGURATION ω_{SP} = 3.9 ζ_{SP} = 0.54 -- / 0.25 / --	FLIGHT/PILOT 1901/B
PILOT RATING: OVERALL --	APPROACH 3 PIO 1	SP 4
WIND/X-WIND: -05/06	TURBULENCE: None	$M_{\delta_{ES}}$ = 0.85

FEEL:
- Forces: Good.
- Displacement: About right.
- Sensitivity: Appropriate.

PITCH ATTITUDE RESPONSE:
- Initial: Quite prompt.
- Predictability: Good.
- Special Inputs: None.
- PIO Tendency: None.

AIRSPEED CONTROL: Easy, tended to fly slow.

PERFORMANCE:
- *Approach Tasks:*
  ILS: Quite good in smooth air.

  Visual (Sidestep): Equally good, easy; could fly more aggressively than normal.
- Landing tasks: Not done.
- Differences: Not applicable.

WIND AND TURBULENCE: Smooth generally but did see a pitch bucking which was undesirable on occasion.

SUMMARY COMMENTS: Suspect that it would be unsatisfactory in moderate turbulence; only problem was turbulence response.

1-6	CONFIGURATION $\omega_{SP} = 1.0$ $\zeta_{SP} = 0.74$ --/ -- /16	FLIGHT/PILOT 1899/A

PILOT RATING: OVERALL --	APPROACH 5	PIO 2	SP 6
WIND/X-WIND: 17/06	TURBULENCE: Moderate		$M_{\delta_{ES}} = 0.26$

FEEL:

- Forces: Moderate
- Displacement: No comments.
- Sensitivity: Best compromise.

PITCH ATTITUDE RESPONSE:

- Initial: A slight delay.
- Predictability: Impaired by mismatch between initial and subsequent response.
- Special Inputs: Used ramp type inputs.
- PIO Tendency: None but a tendency to overcontrol.

AIRSPEED CONTROL: Not a great deal of difficulty; attention diverted somewhat by pitch problems.

PERFORMANCE:

- Approach Tasks:
  ILS: Worked hardest, reasonable.

  Visual (Sidestep): Reasonable.
- Landing tasks: Not done.
- Differences: No comments.

WIND AND TURBULENCE: Some response to turbulence but not a problem; crosswind from right.

SUMMARY COMMENTS: Major problem is hesitancy to use aggressive inputs due to poor predictability.

2-4	CONFIGURATION $\omega_{SP} = 2.3$ $\zeta_{SP} = 0.57$ --/ 0.5 / --	FLIGHT/PILOT 1889/A

PILOT RATING: OVERALL --	APPROACH 5	PIO 2	SP 3
WIND/X-WIND: 05/03	TURBULENCE: None		$M_{\delta_{ES}} = 0.64$

FEEL:

- Forces: Somewhat heavy, but didn't mind them.
- Displacement: No comments.
- Sensitivity: No comments.

PITCH ATTITUDE RESPONSE:

- Initial: Okay, no noticeable lag.
- Predictability: Not very good. Very difficult to figure aircraft out.
- Special Inputs: No comments.
- PIO Tendency: Tendency to overcontrol, no PIO.

AIRSPEED CONTROL: Had to work at it.

PERFORMANCE:

- Approach Tasks:
  - ILS: Was not outstanding.
  - Visual (Sidestep): Approaches showed a slight tendency to overcontrol.
- Landing tasks: Not done.
- Differences: No comments.

WIND AND TURBULENCE: Not a factor.

SUMMARY COMMENTS: Little confused about this one, lot of distractions during evaluations. Not a solid aircraft but not bad, uncertain of rating.

2-4	CONFIGURATION ω_{SP} = 2.3 ζ_{SP} = 0.57 --/ 0.5 /--	FLIGHT/PILOT 1892/B
PILOT RATING: OVERALL --	APPROACH 3 PIO 1	SP 4
WIND/X-WIND: -05/00	TURBULENCE: Light	$M_{\delta_{ES}}$ = 0.38

FEEL:
- Forces: Okay.
- Displacement: A little large.
- Sensitivity: Okay, detected a little lag.

PITCH ATTITUDE RESPONSE:
- Initial: Initial response was slow.
- Predictability: Good.
- Special Inputs: No special pilot techniques (perhaps a little lead.)
- PIO Tendency: No PIO tendency.

AIRSPEED CONTROL: Fairly good.

PERFORMANCE:
- Approach Tasks:
  - ILS: Was okay (bad pilot performance.)
  - Visual (Sidestep): Visual approaches fine.
- Landing tasks: Not done.
- Differences: Not applicable.

WIND AND TURBULENCE: Smooth air.

SUMMARY COMMENTS: Minor problem was small pitch lag and a little more motion than desired. Generaly a pretty good airplane.

4-4	CONFIGURATION $\omega_{SP} = 2.0$ $\zeta_{SP} = 1.06$ --/ 0.5/--	FLIGHT/PILOT 1894/A

PILOT RATING: OVERALL -- APPROACH 5 PIO 2 SP 3
WIND/X-WIND: 00/05 TURBULENCE: Light $M_{\delta_{ES}} = 0.51$

FEEL:
- Forces: Fairly high but wouldn't want lighter.
- Displacement: No comments.
- Sensitivity: No comments

PITCH ATTITUDE RESPONSE:
- Initial: Lagged.
- Predictability: Wasn't very good but not too bad either.
- Special Inputs: Necessary to try to anticipate, reduce aggressiveness.
- PIO Tendency: Tendency to overcontrol.

AIRSPEED CONTROL: More attention required because of lags in pitch but never bad.

PERFORMANCE:
- Approach Tasks:
  ILS: Able to do it.

  Visual (Sidestep): Approaches went fairly well.
- Landing tasks: Not performed.
- Differences: Not applicable.

WIND AND TURBULENCE: Not a factor.

SUMMARY COMMENTS: No comments.

<table>
<tr><td>1-3</td><td colspan="2">CONFIGURATION
$\omega_{SP} = 1.0$ $\zeta_{SP} = 0.74$
-- / 0.25 / --</td><td>FLIGHT/PILOT
1885/A</td></tr>
<tr><td colspan="2">PILOT RATING: OVERALL --</td><td>APPROACH 6 PIO 2</td><td>SP 6</td></tr>
<tr><td colspan="2">WIND/X-WIND: 0/04</td><td>TURBULENCE: Light</td><td>$M_{\delta_{ES}} = 0.26$</td></tr>
</table>

FEEL:
- Forces: High initial forces, comfortable in steady state.
- Displacement: Normal.
- Sensitivity: No comments.

PITCH ATTITUDE RESPONSE:
- Initial: Initial response sluggish.
- Predictability: Final response not predictable; tendency to overshoot a couple of times.
- Special Inputs: Had to fly with small corrections otherwise easy to overcontrol.
- PIO Tendency: No comments.

AIRSPEED CONTROL: Affected by control technique required by pitch.

PERFORMANCE:
- Approach Tasks:
  - ILS: Fairly good, had to pay attention to pitch.
  - Visual (Sidestep): Easy to overcontrol on final approach.
- Landing tasks: Not performed.
- Differences: Not applicable.

WIND AND TURBULENCE: Not a factor.

SUMMARY COMMENTS: Tendency to overcontrol in pitch. Difficult to make large corrections fast.

1-3	CONFIGURATION $\omega_{SP} = 1.0$ $\zeta_{SP} = 0.74$ -- / 0.25 / --	FLIGHT/PILOT 1898/B

PILOT RATING: OVERALL --	APPROACH 6	PIO 3	SP 5
WIND/X-WIND: 13/15	TURBULENCE: Moderate		$M_{\delta_{ES}} = 0.14$

FEEL:
- Forces: Light initially, then heavy; selected on heavy side.
- Displacement: Felt large, spongy.
- Sensitivity: Desirable level.

PITCH ATTITUDE RESPONSE:
- Initial: Very delayed.
- Predictability: Poor.
- Special Inputs: Keep your gain down and fly slowly like a big airplane.
- PIO Tendency: Definite tendency, tended to oscillate 2 or 3 cycles.

AIRSPEED CONTROL: Bothersome, poor, objectionable.

PERFORMANCE:
- Approach Tasks:
  - ILS: Not too bad, but heavily loaded.
  - Visual (Sidestep): Felt behind the airplane on sidesteps; approaches better than on ILS.
- Landing tasks: Not done.
- Differences: Not applicable.

WIND AND TURBULENCE: Moderate corsswind and turbulence.

SUMMARY COMMENTS: Hard to fly. Very delayed initial response made for a high workload, particularly on the ILS.

3-2	CONFIGURATION $\omega_{SP} = 2.2$ $\zeta_{SP} = 0.25$ -- / 0.1 / --	FLIGHT/PILOT 1887/A
PILOT RATING: OVERALL -- APPROACH 6 PIO 3 SP 5 WIND/X-WIND: 11/10 TURBULENCE: Moderate $M_{\delta_{ES}} = 0.26$		

FEEL:

- Forces: Forces fairly light, greater tendency to overcontrol with heavier forces.
- Displacement: No comments.
- Sensitivity: No comments.

PITCH ATTITUDE RESPONSE:

- Initial: Initial response: some lag but not significant.
- Predictability: Final response predictability was poor.
- Special Inputs: No comments.
- PIO Tendency: No comments.

AIRSPEED CONTROL: Reasonably good.

PERFORMANCE:

- Approach Tasks:
  - ILS: Could correct large errors, some tendency to PIO on breakout.
  - Visual (Sidestep): PIO on side step maneuver. Tendency to PIO on go-around.
- Landing tasks: Not done.
- Differences: Not applicable, but worst in later stage of approach.

WIND AND TURBULENCE: Crosswind requires extra attention.

SUMMARY COMMENTS: Difficult to acquire an attitude quickly and predictably. Tendency to PIO.

3-3	CONFIGURATION $\omega_{SP} = 2.2$ $\zeta_{SP} = 0.25$ --/ 0.25/--	FLIGHT/PILOT 1897/A

PILOT RATING: OVERALL -- APPROACH 7 PIO 3½ SP 7

WIND/X-WIND: 11/10 TURBULENCE: Moderate $M_{\delta_{ES}} = 0.26$

FEEL:
- Forces: Comfortable, perhaps a bit light.
- Displacement: No comments.
- Sensitivity: No comments.

PITCH ATTITUDE RESPONSE:
- Initial: Not matched with response after input.
- Predictability: Poor; aircraft bounces in turbulence plus a tendency to PIO.
- Special Inputs: Had to spend a lot of time on attitude.
- PIO Tendency: Yes, had to work hard.

AIRSPEED CONTROL: Not bad but worked hard.

PERFORMANCE:
- Approach Tasks:
  - ILS: Reasonable but hard work.
  - Visual (Sidestep): In a constant oscillation which was largely pilot induced.
- Landing tasks: Not done.
- Differences: Definite difference between instrument [illegible] visual approaches. More PIO tendency [illegible] visual

WIND AND TURBULENCE: Aircraft responded which compounded difficulties.

SUMMARY COMMENTS: Tendency to bobble and to PIO were major problems.

Vita

Randall M. Enright was born on 18 May 1951 at Scott AFB, Illinois. He graduated from high school in Del Valle, Texas in 1969 and attended the United States Air Force Academy from which he received the degree of Bachelor of Aeronautical Engineering in June 1973. He completed navigator training and received his wings in April 1974, and he completed electronic warfare officer training in October 1974. He served as a B-52G Electronic Warfare Officer instructor/evaluator with the 51st Bombardment Squadron, Seymour Johnson AFB, North Carolina until May 1979. During that time he received the degree of Master of Business Management from Central Michigan University. He entered the School of Engineering, Air Force Institute of Technology, in June 1979. He is presently assigned as the Lead Stability and Control Engineer for the Hypervelocity Missile Program, Air Force Armament Laboratory, Eglin AFB, Florida.

Permanent address: 4721 Groveton Way
St. Louis, Mo. 63128

3-8

DTIC